ATARI ST COMPUTER

Die Fachzeitschrift für ATARI ST- und TT-Anwender

September 92

DM 8,- Ös. 64.- / Sfr. 8.- / Lit. 7500,-



VME-Bus-Grafikkarten

Software

ComBase
DataDiet
Pure Pascal
Scooter PCB
Tempus Word 2.0

Falcon 030

Geheimnis gelüftet
Endlich Details

Bauprojekt

ATARI ST mit mehr als 8 MHz

Falcon 030

Profisystem oder Video-Spiel?

Das Staatsgeheimnis ist gelüftet! Erstmals sind nun technische Details über ATARIs neueste Entwicklung - den Falcon 030 - der Öffentlichkeit zugänglich. Nicht nur das: zur alljährlichen ATARI-Messe in Düsseldorf hat man sich vorgenommen, den neuen Rechner im großen Stil zu präsentieren. „Besonders bunt" soll es laut Aussage von ATARI-Geschäftsführer Alwin Stumpf dabei zugehen. Auch diese Aussage läßt darauf schließen, an welche Zielgruppe der Falcon 030 besonders gerichtet ist. Es geht um Spiele und farbige bzw. musikalische Anwendungen für jedermann - und zwar im heimischen Wohnzimmer!

Schon zur CeBIT war in etwa klar, wie das Gerät aussehen würde: ein Consolen-Computer im 1040er-Design. Zeitgleich kamen bittere Kritiken aus den Reihen der ATARI-Anwender: „Wie kann man in dieser Zeit nur einen Spiele-Computer auf den Markt werfen, wo alle Welt auf DIE Profimaschine wartet?".

Ein Spiele-Computer mit 68030-Prozessor, digitalem Signalprozessor und 8-Kanal-16-Bit-Digital-Sound? Kann man so eine Maschine nur aufgrund des Designs als Spiele-Maschine abtun? Ist es nicht vielmehr so, daß ATARI lediglich versucht, endlich eine Profimaschine in die Wohnstuben zu bekommen? Mit einem Tower-Gehäuse, abgesetzter Tastatur und -zig Steckplätzen wäre das wohl kaum durchzusetzen. So ein Gerät stellt kein Vater dem Sohnemann bzw. der Tochter auf (oder unter) den Schreibtisch. Genau da soll der Falcon 030 aber hin. Das Low-Cost-Home-Computer-Marktsegment ist wesentlich größer als der professionelle EBV/DTP-Markt; schließlich zählen die verkauften Stückzahlen, und nur bei ausreichender Verbreitung des neuen Systems ist auch mit qualitativ hochwertiger Software zu rechnen.

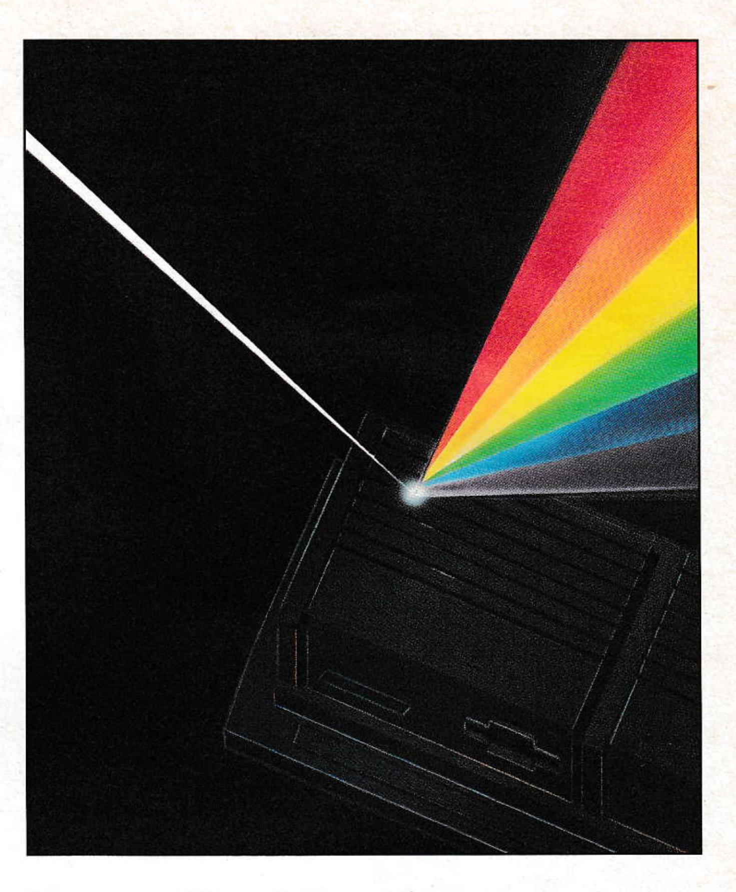
Ein Profi-Falcon-System (mit 68040, VME-Stecklätzen, abgesetzter Tastatur usw. usw ...) wird mit ziemlicher Sicherheit bald folgen. Es kann doch nur positiv für ein solches System sein, wenn es auf einer stabilen Basis aufbauen kann.

CM

INHALT

SOFTWARE

HARDWARE



Bunter Pixel-Strom
VME-Grafikkarten

Atari hat die Rechnermodelle STE und TT nicht gerade mit üppigen Grafikmodi ausgestattet: Vor allem bei Farben und Graustufen zeigen sie sich deutlich unterbelichtet, wie ein Blick nicht nur in die DOS-Welt zeigt. Gleichwohl fand der VME-Bus als Erweiterungs-Port in beiden Rechnern Platz. Naheliegend, daß es mittlerweile ein reichliches Angebot an Grafikkarten für diesen Erweiterungs-Slot gibt. Wir haben uns daher alle verfügbaren Karten herausgesucht, die dem STE/TT eine höhere Grafikauflösung bescheren. Das sind die sechs Modelle COCO (Matrix), Crazy Dots (TKR), Imagine (Wittich), MEGA-Vision (Sang/Trade iT), Spektrum (Wilhelm) und TC1208 (Matrix).

Seite 24

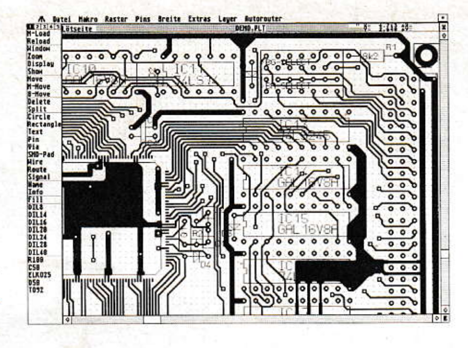
Das Tune-Up-Projekt
ATARI (Mega) ST mit mehr als 8 MHz

Oft wurde dieses Thema diskutiert, aber niemand wollte es so recht für möglich halten. Hier ist der Beweis: der ATARI-ST läuft nach einigen Modifikationen nicht mehr mit 8 MHz, sondern mit 10 oder ggf. mehr und ist auch bei allen Operationen außer Disk-Zugriff entsprechend schneller. Es ist alles vorbereitet, um ein bis zwei HD-Laufwerke anzuschließen, Betrieb mit normalen Laufwerken und gemischt ist aber natürlich auch möglich. Eine sehr einfache, aber wirkungsvolle Bildschirmvergrößerung fällt ebenfalls 'als Abfallprodukt' an. Ein Blick in unser Selbstbauprojekt lohnt sich für Bastler also allemal.

Seite 118

Scooter-PCB

Neues Outfit für Ultimade-PCB



Das in Insider-Kreisen recht renommierte Platinen-Layout-Programm Ultimade-PCB hat einen Nachfolger bekommen. Er nennt sich Scooter-PCB und wird vom gleichen Hersteller vertrieben. Der Vorgänger zeichnete sich besonders durch sein enormes Preis-Leistungsverhältnis aus. Soviel soll schon verraten werden. Was hier mit Scooter-PCB angeboten wird, dürfte so ziemlich jeden Konkurrenten, und nicht nur im ST-Bereich, ins Schwitzen bringen...

Seite 35



Falcon 030

Erste offizielle Leistungsdaten

ATARI hatte sich einen Bilderbuchsommertag ausgesucht, um am 8.7.1992 im Dorint-Hotel in Franfurt Niederrad (kaum 15 Autominuten von unserer Redaktion entfernt) erste offizielle Details des neuen Falcon 030 der Fachpresse mitzuteilen. Bis zu diesem Zeitpunkt war die einzige Information, die ATARI zu diesem Thema preiszugeben bereit war, der lapidare Spruch „no details", der sich in der Szene aus Presse, Entwicklern und Händlern schon zu einem Slogan entwickelt hatte. Nun sollte es also ans Eingemachte gehen. In der Tat war der Zeitpunkt dieser Pressekonferenz klug gewählt. Dadurch sind wir nun in der Lage unseren Lesern pünktlich zur ATARI-Messe die technischen Daten des Falcon 030 zu präsentieren.

Seite 16

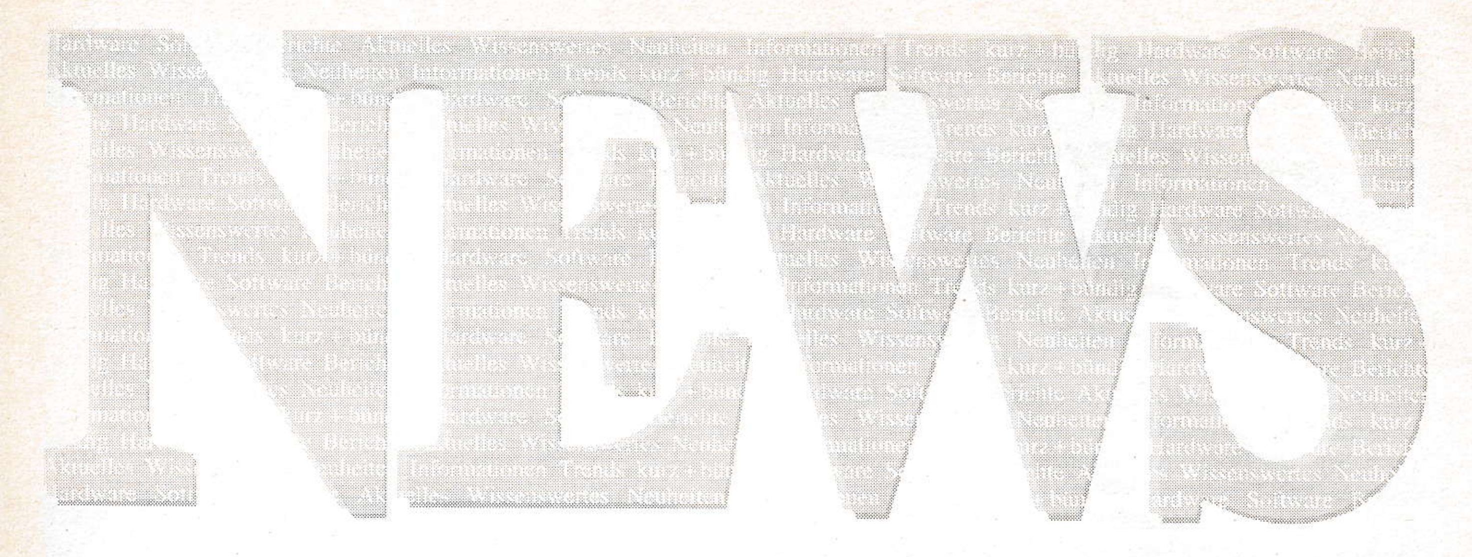
GRUNDLAGEN

PROGRAMMIERPRAXIS

PUBLIC DOMAIN

AKTUELLES

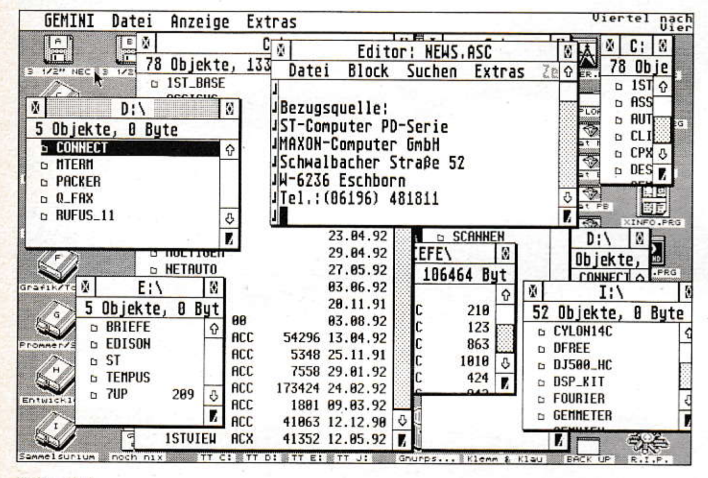
RUBRIKEN



Winx 2.0

Das bekannte Programm Winx, welches den ATARI auf bis zu 128 GEM-Fenster erweitert ist nun in einer brandneuen Version 2.0 verfügbar. Ähnlich wie beim MultiTOS lassen sich jetzt die Rahmenelemente (Verschiebebalken, Scroll-Pfeile, Slider, Close-Button etc.) der Fenster bedienen auch wenn diese nicht im Vordergrund liegen. Dadurch kann man beispielsweise Fenster verschieben, oder vergrößern/verkleinern, ohne sie vorher zu aktivieren. Die Anzahl der Fenster wurde auf maximal 40 begrenzt (mehr wäre wohl kaum sinnvoll). Winx 2.0 kann bequem über ein CPX-Modul konfiguriert werden und läuft mit allen bekannten TOS-Versionen auf ATARI ST-/STE- und TT-Computern.

ST-Computer PD-542
MAXON-Computer GmbH
Schwalbacher Straße 52
W-6236 Eschborn
Tel.:(06196) 481811



Winx 2.0

STARTER-Update

Starter, das Accessory zum bequemen Starten von Programmen auf Funktionstastendruck, liegt nun in der stark verbesserten Version 4.1 vor. Neben den Grow- und Shrink-Boxen ist auch der komplette Bildschirmaufbau beim Programmstart abschaltbar, woraus eine deutliche Geschwindigkeitssteigerung resultiert. Zusätzlich wurde durch ein abschaltbares AV-Protokoll die Kompatibilität erhöht. Geblieben sind die einfache Bedienung sowie die problemlose Zusammenarbeit mit dem Atari-Desktop und der Shareware-Shell GEMINI. Ideal für Backups und DFÜ ist die Möglichkeit, einzelne Programme automatisch zu bestimmten Uhrzeiten zu starten. Der Preis beträgt weiterhin 35,- DM. Ein Update ist für 15,- DM erhältlich.

Bernd Blank
Friedrichstaler Allee 6
W-7513 Stutensee 4
Tel.:(07249) 4422

Font-Katalog zu Calligrapher

Von der Firma Working Title wird die Fertigstellung des Calligrapher-Font-Kataloges gemeldet. Er beschreibt die 64 zusätzlichen Vektor-Fonts in Schriftbeispielen, gibt jeweils Hinweise auf den Fonttyp (empfohlene Größen, Textkörper, Headline, Kapitalschrift usw.) und einen Überblick über die zu den Fonts gehörenden Schnitte. Da die Fonts von URW Hamburg lizensiert sind und den PostScript-Type-1-Fonts entsprechen, können sie über den PostScript-Treiber von Calligrapher professional auf entsprechende Drucker und Satzbelichter ausgegeben werden. Der Font-Prospekt kann gegen Einsendung von 3,- DM in Briefmarken angefordert werden bei:

Working Title GbR
Lilienweg 12
W-5300 Bonn 1
Tel.: (0228) 647020

Endlich: Formeln im Text

FORMEL-X heißt ein Programm, das das leidige Problem vieler Textverarbeitungsprogramme löst, die z.B. keine mathematischen Formeln darstellen können. Die Formel wird extern erstellt und als IMG-Grafik (maximal 1920*600 Bildpunkte) in die Druckerauflösung gewandelt, welche dann die Textverarbeitung einlesen kann. FORMEL-X ist als Accessory lauffähig, hat einen eigenen Editor mit Sofortformatierung und WYSIWYG. Preis: 149,- DM.

HEIM Verlag
Heidelberger Landstraße 194
W-6100 Darmstadt
Tel.: (06151) 56057

TETRA meldet sich zurück

Nach einer längeren Abstinenz vom ATARI-Markt meldet sich die Firma TETRA Computersysteme jetzt mit einem TT-Tower zurück.

Zum Preis von 2445,- DM kann jeder TT-Besitzer seinen Rechner in einen „TETRA 030" verwandeln lassen. Natürlich ist der TETRA 030 auch als komplett ausgestatteter Tower-TT (2 MByte ST-RAM, 8 MByte TT-RAM, Festplatte Quantum 240 MByte 15ms SCSI, ohne Monitor) in der Basisversion zum Preis von 7885,- DM erhältlich. TETRA gewährt auf das Gehäuse mit Einbau bzw. das Komplettsystem eine Garantie von 24 Monaten!

TETRA Computersysteme GmbH
Neuer Markt 27
W-5309 Meckenheim
Tel.: (02225) 17081

Interface 2.0

Die Firma Shift bietet das bekannte Resource-Construction-Set Interface in der Version 2.0 an. Einige der zahlreichen Neuerungen hier im Überblick:

Alle Dialoge wurden in Fenster gelegt (abschaltbar!). Zwei Resource-Dateien können nun miteinander verglichen werden. Sinn der Funktion ist, daß man damit z.B. fremdsprachige Resourcen leichter pflegen kann und Fehler schneller findet. Die Baum- und Objektebox ist in ein Fenster gelegt worden und muß daher nicht immer erneut über „O“ oder „B“ aufgerufen werden (abschaltbar). Man kann nun in der Suchfunktion auch nach irgendwelchen Texten in der Resource suchen. Groß/Kleinschreibung wird unterschieden. Resourcen dürfen jetzt (fast) beliebig lang werden. Die Anzahl der Objekte pro Baum ist auf 1024 begrenzt; jede Resource kann unbegrenzt viele Bäume enthalten (kein Beschränkung mehr auf max. 256 Bäume). Wenn mehr als 256 Bäume verwendet werden, muß die Resource im HRD-Format abgespeichert werden. Routinen zum Laden solcher Resourcen kann jeder im C-Source erhalten. Die Verwaltung ist 100% kompatibel zu den alten Resourcen.

Im Einstellungsdialog kann man eine Sizebox einschalten, die für jedes Objekt unter der Maus die Size-Ecke anzeigt. Für alle nützlich, die die Ecke nie finden konnten. Man kann jetzt eigene Objekte in die Objekte-Box von Interface einbinden (FlyDial-Ecke, OK-Button, ...). Dazu lädt man die Interface-Resource und kopiert das gewünschte Objekt in den „OBJEKTE“-Baum. Die Sortierreihenfolge ist egal, es kommt nur darauf an, daß das neue Objekt in der inneren Box landet, in der sich auch die anderen Objekte befinden. Ein Clipboard für Objekte ist jetzt eingebaut. Objekte zieht man einfach aufs Clipboard bzw. zieht sie wieder aus dem Clipboard auf ein Dialog- oder Menüfenster. Wenn das Clipboard-Icon verschoben werden soll, muß man es erst selektieren (Mausklick oder Rubberbox) und dann verschieben. Solange es deselektiert ist, kann man den Inhalt „herausziehen“. Man darf selbstverständlich auch mehrere Objekte gleichzeitig aufs Clipboard kopieren. Unter MultiTOS kann man jetzt auch in den Toolbox-Fenstern und im Fenster-Desktop arbeiten, wenn sie im Hintergrund liegen (sie werden nicht getoppt). Man muß dann dabei nicht mehr die rechte Maustaste drücken.

Dies sind längst nicht alle Neuerungen. Auf der ATARI-Messe wird Interface 2.0 vorgestellt.

Shift
Unterer Lautrupweg 8
W-2390 Flensburg
Tel.:(0461) 22828

Kopieren ohne Computer

Mit dem SOLO-Disk-Kopierer ist es nun erstmals möglich, Disketten mit bis zu 1.44 MB Speicherkapazität auch ohne Computer zu kopieren. Der Kopiervorgang dauert zwischen 30 und 60 Sekunden, je nach Format. Benötigt werden lediglich zwei Diskettenlaufwerke, zwischen die SOLO geschaltet wird. Auf Knopfdruck kann man dann ein- und doppelseitige Kopien erstellen. SOLO kopiert sovielе Spuren, wie es ihr Laufwerk zuläßt. Für den Fall, daß man SOLO zwischen einen Computer und ein zweites Laufwerk schalten will, wird ein Programm mitgeliefert, das den Computer steuert. Der Computer wird durch den Kopiervorgang nicht belegt, so daß man dabei weiterarbeiten kann.

Cachet
Ostendstr. 32
W-7524 Östringen

EAN Barcode-Reader-System

Das Barcode-Reader-System besteht aus einem hochqualitativen Barcode-Lesestift sowie einem Barcode-Adapter zur Umsetzung der Barcode-Daten in den Rechner. Der Adapter kann auf die Mega-ST/E- oder TT-Tastatur aufgesetzt werden und wird in die serielle Schnittstelle eingesteckt. Das im Lieferumfang enthaltene Leseprogramm übergibt die gelesenen Daten in den Tastaturpuffer Ihres Computers. Auf diese Art ist es möglich, auch Programme zu bedienen, die keine Barcode-Möglichkeit anbieten. Optimal bei selbsterstellten Datenbankanwendungen oder Programmen, die mit einer Artikelnummer oder ähnlichem arbeiten.



Das Paket besteht aus dem Barcode-Lesestift, dem Barcode-Adapter, sämtlichen Anschlußkabeln, dem EAN Strichcode Generator Professional sowie dem neuen Barcode Transmitter Programm. Preis komplett 1299,- DM. Der EAN-Strichcode-Generator Professional ist das optimale Tool für Grafiker, die einen Strichcode in ein Verpackungs-Layout integrieren wollen. Bislang wurde der Strichcode noch nachträglich von Hand einmontiert oder als Film gekauft. Mit dem EAN-Strichcode-Generator erstellen Sie den Strichcode als GEM-IMG und importieren ihn direkt in Ihr DTP-Programm, um ihn dort beliebig zu setzen. Der Strichcode Generator erstellt sechs verschiedene Strichcodes, die Sie hier verkleinert als Beispiel abgebildet sehen können. Die Strichcodes entsprechen in Form und Größe den Vorgaben der Centrale für Coorganisation (CCG), welche auch die Strichcodes vergibt. Der EAN-Strichcode-Generator ist auch als Accessory lauffähig. Damit haben Sie die Möglichkeit, den Strichcode-Generator in Ihrem DTP- oder Grafikprogramm aufzurufen. Die Ausgabe des Strichcodes erfolgt hierbei direkt aus dem verwendeten Grafik- oder DTP-Programm auf 24-Nadel-, Laserdrukker oder Satzbelichter. Barcodes im DTP Programm - ideal bei der Gestaltung von Verpackungen, Buchhüllen und ähnlichem.
In der PRG-Version besteht auch noch die Möglichkeit, Barcodes direkt auszudrucken. Hierbei lassen sich die Codehöhe einstellen sowie Spaltenbreite und -höhe für Etiketten eingeben. Der Ausdruck erfolgt über einen 24-Nadel oder den SLM-Laserdrucker. Hier können Endlosetiketten oder, beim Laserdrucker, Laseretiketten verwendet werden. Wenn Sie die Codes aus einem Warenwirtschaftssystem als ASCII exportieren oder als ASCII-Text eingeben, ist es auch möglich, verschiedene Codetypen nacheinander mit Zusatztexten direkt auszudrucken, auch auf mehrspaltigen Etiketten. Auch der Endlosausdruck eines schon als IMG-File generierten Codes ist möglich. Das direkte Ausdrucken ist ideal zum Etikettieren im Ladengeschäft oder für die Produktion.

Eickmann-Computer
In der Römerstadt 249/253/259
W-6000 Frankfurt 90
Tel.:(069) 763409

ARTIS bekennt Farbe!

ARTIS, das Grafikprogramm aus Österreich, ist in der Version 3.0 voll farbfähig geworden. Von Monochrom bis True Color ist alles machbar, was ATARI-Monitore darstellen können. Gleichzeitig unterstützt ARTIS alle (!)SIGNUM!-Fonts, die neuen FMS-GDOS-Vektorschriften und läuft selbstverständlich unter MultiGEM, MultiTOS sowie auf dem Falcon030. Das GEM-Arbeitsfenster ist völlig neu gestaltet worden und ist deshalb noch übersichtlicher. ARTIS 3 kostet 398,- DM.

ARTIS Software
W. Ante & A. Beza G.n.b.R.
Hohlweggasse 40 - 54
A-1030 Wien

Chemplot jetzt in Farbe

Der bekannte Strukturformeleditor für alle Chemiker CHEMPLOT ist nunmehr in der Version 2.1 erhältlich. Alle Chemplot-Objekte können jetzt farblich gestaltet werden, was die Übersichtlichkeit wesentlich erhöht. Außerdem arbeitet das Programm inzwischen völlig auflösungsunabhängig. Neu ist auch eine Strukturformel-Datenbank, in der alle Molekülstrukturbilder mit Erläuterungstext Platz finden. Die GEM-konforme Oberfläche wurde überarbeitet und bietet verschiebbare Icon-Leisten und Pop-Up-Menüs. Die neue Undo-Option macht versehentlich erfolgte Arbeiten wieder ungeschehen. Jetzt kann auch die Zeichengröße für Ladungen und Radikale geändert werden. CHEMPLOT kostet neu 148,- DM, ein Update-Service wird angeboten.

Chemo-Soft
Jürgen Osterthun
Lindenhofsgarten 1
W-2900 Oldenburg
Tel.: (0441) 82851

Neues von PAMs NET

Rechtzeitig zur ATARI-Messe ist eine neue Version des Netzwerkprogramms für ST/TT fertiggstellt worden. Es ist nunmehr möglich, mehrere Netzwerke untereinander zu verknüpfen und sogar Einzelplatzrechner per Modem (!) zu integrieren. Dabei wird der Einzelrechner einfach mit der seriellen Schnittstelle eines beliebigen Netzwerkteilnehmers, der eine NET/VMC-Karte besitzt, verbunden. Des weiteren ist das Modem, unterstützt durch die Programme QFax/net, Teleoffice o.ä., als zentraler FAX-Server einsetzbar. Eine Einbindung in Standard-NOVELL-Netze und der Zugriff per NFS auf UNIX-File-Systeme ist ebenfalls möglich.

PAM Software GmbH
Pascal Merle
Carl Zuckmayer Straße 27
W-6500 Mainz
Tel.: (06131) 476312

Neues Netz

Der Markt für die Vernetzung von Einzelarbeitsplätzen (PC) wird nach Prognosen in den nächsten Jahren um ein Mehrfaches steigen. Auch die ATARI-Rechner sind mit von der Partie. Neu in der Runde der ATARI-Netze ist ANS - Advanced Network System, das die Firma Wacker Systemelektronik ab sofort vertreibt.
Die Verwendung von standardisierten Protokollen wie Ethernet in Verbindung mit TCP/IP garantieren ein hohes MAß an Zugangssicherheit zu anderen Systemen. Die aktuelle Version 3.05 bietet einen „Non/Dedicated"-Betrieb, ist voll Internet-kompatibel, erlaubt Multitasking u.v.m.

wacker systemelektronik gmbh
Honsellstraße 1 a-d
W-7500 Karlsruhe 21
Tel.: (0721) 95560-0

Schaltplan-Design

STANED heißt ein Programm, mit dem man Schaltpläne entwerfen kann, und zwar nicht nur Einzelblatt-, sondern auch Multilayer-Pläne. Ein Symboleditor ist fester Bestandteil des Programms. Es läuft voll unter GEM auf allen ST/STE/TT in Auflösungsstufen ab 640*200 Bildpunkten. Die Größe der Schaltpläne und die Anzahl der parallel offenen Arbeitsfenster wird nur durch die Kapazität des Arbeitsspeichers begrenzt. Treiber für 9-, 24-Nadel- und HP-Laserdrucker sind mit dabei. Eine zusätzliche Ausgabe als IMG-File gibt es auch.
Als Besonderheit können aus den Schaltplänen sog. Netzlisten für verschiedene Layout-Programme, für das Schaltungssimulationsprogramm SPICE sowie für den Logic-Simulator STANLOG erzeugt werden. Ab 1. September ist das Programm zum Einführungspreis von 149,- DM im Handel

BCP-Hard&Soft
Dirk Lehmann
Im Dorfe 19

Indexus II

Jäger und Sammler

Für die meisten Computernutzer ist sie vorbei, jene Zeit, als sie noch wie Diskjockeys ständig von Programm- auf Datendisketten und umgekehrt unsteigen durften. Sie erinnern sich? Ganz schnell schleichte sich beim Jonglieren dieser Massenspeicherscheiben (schönes Wort - gelle?) das Gefühl der Unvollkommenheit ein, weil es einem ständig auf den Keks ging, immer nach jener Diskette suchen zu müssen, die man gerade brauchte und selten sofort fand (Murphy hat ein Gesetz dafür).

Warum sich also durch riesige Diskettenstapel auf dem Schreibtisch arbeiten (übrigens: unter „Stapelverarbeitung" ist wahrhaftig etwas anderes gemeint), wenn es etwas gibt, das uns all die Wühlarbeit abnimmt und zudem nach schneller ist? Langer Rede kurzer Sinn: Festplatten müssen es sein!

Dann wird ganz schnell der gesamte Programmbestand auf eine solche Harddisk kopiert und ab sofort kennt der Anwender nur noch einen Handgriff: das Betätigen des Netzschalters (der Glückliche). Ob Sie's nun glauben oder nicht, es gibt auch Murphys Gesetze für Festplatten, Beispiel: Eine Festplatte ist niemals groß genug - gleichgültig wie groß sie ist.

Angesichts schier unermeßlich großer Speicherkapazitäten und der menschlichen Unvollkommenheit, auch auf Anhieb, trotz nützlicher Grafikoberfläche und Icons den richtigen Festplattenzugriff gefunden zu haben, oder beim Abarbeiten größerer Türme von Sicherheitskopien just die gesuchte Diskette immer als allerletzte hervorzukramen, haben sich weitsichtige Zeitgenossen einiges einfallen lassen, um über die Datenbestände den Überblick zu wahren (und zu behalten). Programme zur Verwaltung der Speicherbestände sind MEGA-in, denn ein anderes Gesetz laut Murphy besagt, daß Datenbestände die Unart haben, unaufhörlich zu wachsen.

Verwaltungschef

Aus den Hause MGL-Soft liegt uns schon seit geraumer Zeit eine Software vor, die genau bei diesem Problem in die Bresche springt - Name: Indexus II professional - spezialisiert auf Dateibezeichnungen aller Art von Diskette, Fest- und Wechselplatte. Im Grunde könnte man ja alles mit einem simplen Datenbankprogramm lösen. Man konstruiert sich sinnvolle Masken und tippt brav, dem alten Karteikastenprinzip folgend, die Programmnamen ein. Aber einmal ehrlich, ist das nicht oberumständlich?

INDEXUS hätte keinen Sinn, wenn es nur eine spezialisierte Datenbank wäre (MEGA-out), vielmehr ist es ein Inhaltsverzeichnisleseprogramm das es IN sich hat (im wahrsten Sinne des Wortes). Man könnte die Aufgabenstellung von INDEXUS folgendermaßen umschreiben:

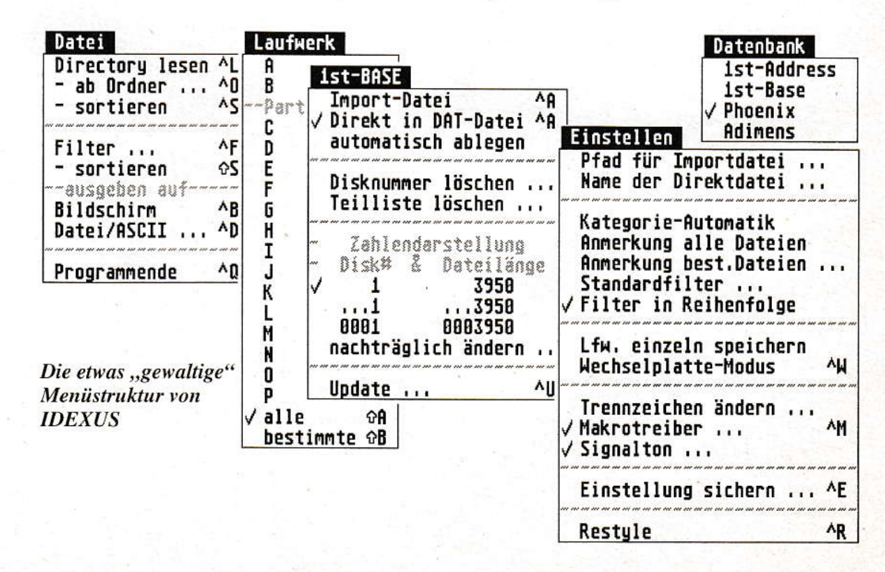
Wenn wir davon ausgehen, daß fast jeder Computernutzer auch ein Datenbankprogramm besitzt, macht es keinen Sinn, ebenfalls eine Datenbank zu sein. Lassen wir das schnelle Suchen und Sortieren doch lieber später von einer Datenbank erledigen (dazu haben wir sie ja). Besinnen wir uns auf die eigentliche Aufgabe (und dafür gibt es herzlich wenig Auswahl auf dem Softwaremarkt), Inhaltsverzeichnisse automatisch einzulesen, dabei den Disketten (oder Wechselplatten) auch gleich eine Registrier- oder Ordnungsnummer zu vergeben, nach Dateierweiterungen (Extensions) zu sortieren, spezielle Extensions anzuzeigen oder zu unterdrükken, mit einer Update-Funktion Änderungen zu brücksichtigen ... was, das reicht Ihnen noch nicht - warten Sie einmal ab!

Die Guten ins Töpfchen ...

INDEXUS liest automatisch sämtliche Inhaltsverzeichnisse einer Festplatte, oder von Disketten- und Wechselplattenmedienfolgen nacheinander ein. Dabei vergibt das Programm gleichzeitig eine Kennung, bei der Festplatte bleiben die Buchstaben der logischen Laufwerke, bei Disketten gibt es eine Nummer von 0000 bis 9999, bei Wechselplatten steht der Nummer noch ein „W" voran.

Natürlich werden die sogenannten Extensions besonders berücksichtigt. Mittels einer Filterfunktion kann man beim Einlesen nur bestimmte Dateierweiterungen berücksichtigen (weil einem halt andere, wie „XYZ", „JWD" oder „123" partout nicht interessieren). Selbstverständlich lassen sich die Extensions welche den Filter passiert haben, noch alphabetisch sortieren.

Sind auf diese Weise nun alle „Datenaußenstände" erfaßt worden, zeigt das Programm diese in einem speziellen „Aufbereitungsfenster" an. Im Handbuch wird nicht verschwiegen, daß dieses Aufbereitungsfenster nur sehr unvollkommen ist und für datenbanktypische Arbeiten (suchen, sortieren, scrollen usw.) eigentlich nicht gedacht ist. Es dient vielmehr der Anzeige aller eingelesenen Dateiinforma-



Die etwas „gewaltige" Menüstruktur von IDEXUS

tionen (linkes Fenster), aller bisher vorgekommenen Extensions (rechtes Fenster), sowie der Feststellung, aus welchem Pfad, mit welcher Länge, Datum und Erstellungszeit die Datei „behaftet" war.

Einzige Arbeit die jetzt noch möglich ist, wäre das zusätzliche Vergeben einer Kathegorie und das Anbringen eines Bemerkungstextes pro Datei. „Kathegorie" ist die Grobeinteilung nach dem altbekannten Schema: Textverarbeitung, Datenbank, Tabelle, Spiel usw., während „Anmerkung" ein individueller Text sein kann. Fast selbstverständlich lassen sich die Filterbedingungen auch für die Vergabe dieser Zusatztexte verwenden. Außerdem lassen sich 20 dieser Zusatztexte mit maximal 25 Zeichen als Makros auf die Funktionstasten legen, das erspart Tipparbeit bei häufig vorkommenden Texten.

Zusammenarbeit

Dateien die INDEXUS zusammengestellt hat, lassen ich als sog. „Direktdatei" sofort in den Datenbankprogrammen „1st-BASE" und „1ST-ADRESS" in Listenform weiterverwenden. In 1st-BASE (als Accessory installiert) kann man sofort auf diese Direktdatei zugreifen und alle datenbanktypischen Arbeiten ausführen. Weil die Direktdatei bei 1ST-ADRESS neue Daten hinter eine bestehende Datei hängt, ist hier (trotz ACC-Installation) erst ein Reset nötig. Besonderes Bonbon dabei: Weil INDEXUS schon fertige Datendateien für diese beiden Datenbanken anbietet, können auch deren Demoversionen vollwertig benutzt werden.

Auf weitere gute Zusammenarbeit baut INDEXUS mit den Programmen ADIMENS und PHOENIX. Dort gestaltet sich die Vorgehensweise etwas anders, weil zur Darstellung von Daten spezielle Masken konstruiert sein müssen. Auch hier null problemo: INDEXUS liefert auf Diskette vorbereitete Masken- und Indexdateien mit.

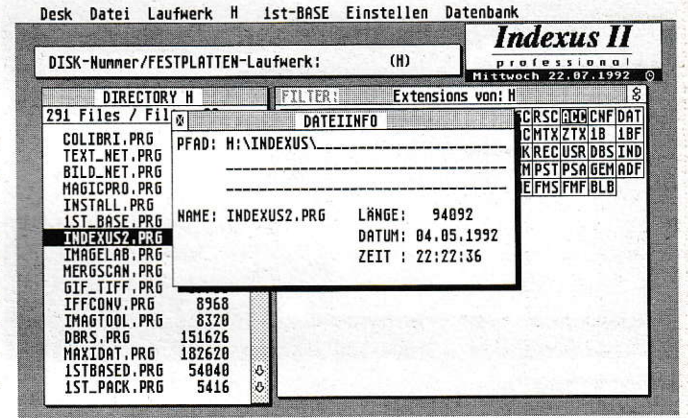
Selbstverständlich kann man sich auch eingene Masken basteln und die Katalogdaten per ASCII importieren.

Up to date

Etwas besonderes stellt die Update-Funktion dar. Immer dann, wenn sich in der Zusammenstellung eines Datenträgers etwas ändert, wäre es zu stupide, immer den Gesamtbestand neu abzufragen (wäre auch zu einfach). Die Update-Funktion erfragt jetzt nur die „Neuzugänge" an Dateien ab und aktualisiert die zugehörigen INDEXUS-Kataloge.

Vereinfachung

Es wird wohl kaum jemanden geben, der seine Diskettensammlung manuell mit einem Datenbankprogramm „betreut" (back to the Karteikasten). Sicherlich gibt es auch Directory-Verwalter im PD oder Shareware-Bereich. Trotzdem stößt INDEXUS in eine Markt- bzw. Anwendungslücke, weil es die sinnvolle Verbindung, quasi eine „Durchreiche" zu vier bekannten Datenbankprogrammen gibt. Und gerade deshalb kann man in INDEXUS selbst gerne auf datenbanktypische Funktionen verzichten.



Die INDEXUS-Oberfläche, nicht ganz GEM-like

Zielgruppe dieses Programms ist wohl nicht so sehr der Hobbyanwender, wenngleich es auch für ihn zahlreiche Anwendungsfälle gibt. Sinn macht das Programm viel eher bei semiprofessionellen Einsätzen, wo man einfach nicht die Zeit hat, dutzende von Disketten nach der gesuchten Bilddatei zu durchforsten. Und gerade dann, wenn „Zeit = Geld" ist, macht sich INDEXUS bezahlt.

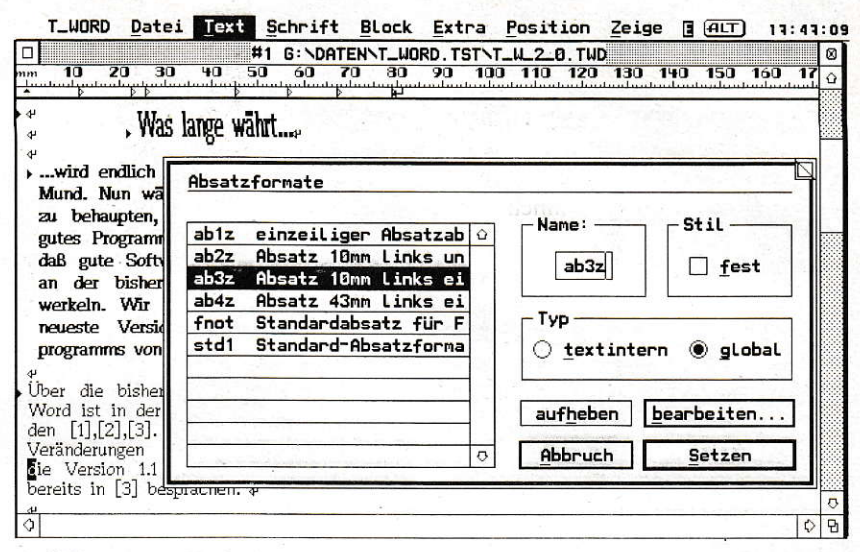
Sehr gut gefallen hat mir das Handbuch, das auf 72 Seiten (DIN-A5) sehr ausführlich alle Funktionen des Programms und die Zusammenarbeit mit den vier Datenbanken erläutert. An Bildern, sowie Tips und Tricks wurde nicht gespart. Mit 99 DM (zzgl. 10 DM für Versand) liegt das Programm gerade noch im Rahmen des tolerierbaren.

DK

Bezugsquelle:
MGL-Soft
Haidestraße 7
A-4600 Wels
Tel.: 0043/7242/26418

Tempus Word 2.0

Was lange währt...



Auffällig: Tempus Word mit neuer Benutzerführung

...wird endlich gut. So sagt es des Volkes Mund. Nun wäre es sicherlich vermessen zu behaupten, Tempus Word wäre kein gutes Programm. Aber getreu dem Motto, daß gute Software nie fertig ist, gab es an der bisherigen Version 1.1 noch zu werkeln. Wir freuen uns, Ihnen hier die neueste Version des Textverarbeitungsprogramms von CCD vorstellen zu können.

Über die bisherigen Versionen von Tempus Word ist in der ST-Computer berichtet worden [1],[2],[3]. Im folgenden soll auf die Veränderungen eingegangen werden, denen die Version 1.1 unterworfen wurde, die wir bereits in [3] besprachen.

Bereits bei der Installation der neuen Version fällt auf, daß die Programmlänge gesunken ist. Dadurch und durch klügere Verwaltung des Speicherplatzes braucht Tempus Word nun 100 KB weniger Speicher, was gerade Benutzer ‚kleiner' ST-Systeme erfreuen dürfte. Wer hingegen einen TT sein eigen nennt, hatte bislang mit dem äußerst lästigen 24-Bit-Problem zu kämpfen. Das gibt es nicht mehr, das Programm ist uneingeschränkt auf dem 68030 lauffähig.

Viele Operationen sind in ihrer Ausführungsgeschwindigkeit gestiegen, so speichert Tempus Word die Texte jetzt bis zu 7mal schneller. Die weitere Erhöhung der Performance beruht auf der neuen Dialogführung, die jetzt eigene Routinen zur Darstellung benutzt und durch drastisch schnelleren Bildaufbau glänzt. Davon (und von den Korrekturvorschlägen) abgesehen, ließ Tempus Word ohnehin nie große Wartezeiten zu.

Human Interface

Die sicherlich auffälligste Änderung an Tempus Word 2.0 betrifft die Dialogführung. Sie ist vollkommen auf ‚fliegende Dialoge' umgestellt und wesentlich vereinfacht worden. Die Dialoge und die Menüleiste sind fast vollständig mit der Maus bedienbar, die Oberfläche entspricht damit dem aktuellen GEM-Standard. Die Tastatursteuerung erstreckt sich ebenfalls auf den eigenen File-Selector, den Tempus Word mitbringt. Er ist damit so flüssig bedienbar geworden, daß CCD ihn ruhig per Auto-Programm ins Betriebssystem einklinken könnten.

Die besonders mit Parametern überfrachteten Dialoge wie die Absatzgestaltung oder der Drucken-Dialog wurden entwirrt und sind jetzt auch ohne Übung einigermaßen schlüssig. Der Nebeneffekt der neuen Dialoge ist die Makrofähigkeit von Tempus Word. Jede Aktion ist per Tastenfolge erreichbar und kann aufgezeichnet, gespeichert und zu späterem Zeitpunkt wieder abgespielt werden. Leider ist das Editieren von Makros nicht möglich. Wem die Tastaturbelegung von Tempus Word nicht gefällt, der kann sich ‚Text öffnen' auch auf <Control-L> legen, indem er die entsprechende ASCII-Steuerdatei ändert. In dieser sind sämtlichen Befehlen des Programms beliebige Tastenkombinationen zuzuordnen.

Durch diese Neuerungen - Umbelegung der Tastatur und Makros - läßt sich Tempus Word jetzt an eine Unzahl Situationen anpassen. Wiederkehrende Tätigkeiten werden damit zum Kinderspiel.

Zusätzlich steht jetzt ein Font-Modus zur Verfügung, bei dem die Tastaturkommandos zwar nicht mehr zugänglich sind, aber stattdessen andere Fonts auf die Control- und Alt-Ebene gelegt werden können. Verfassern fremdsprachiger Texte wird das entgegenkommen.

Bei der Arbeit...

...fallen die überarbeiteten Blockfunktionen auf, deren Fehler beseitigt wurden. Außerdem ist das Aufziehen eines Blokkes mit der Maus jetzt endlich möglich. Eine Unzahl von Details ist geändert (und meist verbessert) worden.

Angenehm fallen z.B. die verbesserte Dialogführung bei der Korrektur oder einfach nur der zentrierte Tabulator und die Lote auf, die beim Verschieben eines Tabs erscheinen. Der Dateimanager beherrscht jetzt auch das rekursive Durchsuchen ganzer Partitionen nach allen oder nur bestimmten Texten. Es fehlt Tempus Word leider noch immer der Formelsatz à la TEX und ein Tabellensatz, wie ihn Cypress vormacht.

Gratwanderung

Wo ist die Grenze zwischen DTP und Textprogramm? Die Beantwortung dieser Frage wird zunehmend schwieriger. Während die Textbearbeitung mit DTP-Programmen zumindest möglich ist, erhalten

Textverarbeitungen mehr und mehr Möglichkeiten zur Gestaltung der Dokumente. So hat Tempus Word mächtig dazugelernt, was die Verarbeitung von Bildern betrifft. Nunmehr ist auch das Laden von Vektorgrafiken (im *.GEM-Format) möglich, diese können genauso wie die Pixel-Bilder frei im Text verschoben und skaliert werden. Das geht ähnlich wie bei DTP-Systemen mit der Maus.

Darüber hinaus stehen jetzt grafische Objekte wie Linien, Rechtecke und Kreise zur Verfügung, mit denen man Hervorhebungen o.ä. realisieren kann, ohne extra ein Grafikprogramm zu bemühen und das Bild einzuladen.

Output

Das Ziel der Textverarbeitung ist nach wie vor die Ausgabe auf einem Stück Papier. Auch in diesem Bereich hat Tempus Word noch einiges dazugelernt. Es können jetzt beliebige Seitengruppen an den Drucker übergeben werden. Dabei wurde die gesamte Handhabung der Druckerwarteschlange und des Spoolers verbessert. Während es immer noch die Möglichkeit gibt, zeitsparend im Textmodus die Daten in reiner ASCII-Form an den Drucker zu schicken, gibt es jetzt noch ein Zwischending für HP-kompatible Laserdrucker. Dabei werden zwar die grafischen Zeichensätze von Tempus Word benutzt, aber trotzdem wesentlich weniger Daten geschickt. Gibt man hingegen Grafik auf einem HP-Laser aus, sendet Tempus Word die Daten jetzt komprimiert, was wiederum zu einer erheblichen Reduzierung der Menge führt. Wessen Drucker Face-up ausgibt, der kann die Option ‚rückwärts‘ wählen und erhält das Dokument richtig sortiert.

Der Druck auf den Monitor erlaubt die pixelgenaue Kontrolle des Druckbildes; sozusagen als Abfallprodukt kann Tempus Word jetzt auch das Druckergebnis als GEM-IMG-Rastergrafik speichern. Auf 66 Prozent verkleinert (auch das ist möglich!) kann so das Blatt mit einem 200-DPI-Fax versandt werden. Doch dessen nicht genug: Tempus Word kann jetzt auch direkt faxen. Mit den entsprechenden Treibern von QFAX oder TeleOffice im Hintergrund wählt man ‚Ausgabe FAX‘, und schon startet die Fernkopie (ein entsprechendes Modem vorausgesetzt).

Weiterhin neu ist der Querdruck, der besonders für breite Tabellen sehr praktisch ist, sowie der Mutterdruck, bei dem eine DIN-A5-Seite zweimal nebeneinander auf eine DIN-A4-Seite gedruckt wird. Schneidet man ein ganzes so gedrucktes Dokument auseinander, erhält man zwei DIN-A5-Exemplare und hat kein Papier verschwendet. Last not least ist die Zwischenlagerung der Druckdaten auf der Festplatte möglich, was die Druckzeiten ebenfalls verringern hilft.

Fortschritt

Die Version 2.0 von Tempus Word bietet zwar häufig nur im Detail Verbesserungen, dafür in ganz entscheidenden. Ein Beispiel: Die Seitenvorschau paßt sich jetzt immer an die aktuelle Auflösung an, was Besitzern von Großbildschirmen fast eine 1:1-Darstellung bietet. Früher verlor sich die stark verkleinerte Darstellung auf dem großen Schirm.

Dadurch, vor allem aber durch die entwirrte Benutzerführung und die Makrofähigkeit, hat Tempus Word eine Funktionalität erreicht, wie sie auf dem Textverarbeitungssektor beim ST kaum erreicht ist. Hier und da bieten andere Programme mehr, die Universalität und Geschwindigkeit von Tempus Word sucht aber ihresgleichen.

IB

Literatur:

[1] Moderne Zeiten - Tempus Word Vorabbericht - ST-Computer 6/90 S.38 ff.
[2] Na endlich - Tempus Word ist da. - ST-Computer 2/91 S. 19 ff.
[3] Schreibmaschinen: Sieben Textverarbeitungen für den ST/TT - ST-Computer 4/92 S. 39 ff.

Bezugsadresse:

CCD
Hochheimer Straße 194
W-6228 Eltville
(06123)1638



Falcon 030

NEW CHIPS ON THE BLOCK

Erste offizielle Leistungsdaten

Stahlblauer Himmel, ein leichter angenehmer Wind, und das Thermometer zeigte bereits um 11.00 Uhr ca. 20°C. ATARI hatte sich einen Bilderbuchsommertag ausgesucht, um am 8.7.1992 im Dorint-Hotel in Franfurt Niederrad (kaum 15 Autominuten von unserer Redaktion entfernt) erste offizielle Details des neuen Falcon 030 der Fachpresse mitzuteilen.

Bis zu diesem Zeitpunkt war die einzige Information, die ATARI zu diesem Thema preiszugeben bereit war, der lapidare Spruch „no details“, der sich in der Szene aus Presse, Entwicklern und Händlern schon zu einem Slogan entwickelt hatte. Nun sollte es also ans Eingemachte gehen. Gespannt und voller Ungeduld lauschte man den Worten Normen Kowalewskis, ja, man sog die Informationen richtig in sich auf. In der Tat war der Zeitpunkt dieser Pressekonferenz klug gewählt. Dadurch sind wir nun in der Lage unseren Lesern pünktlich zur ATARI-Messe die technischen Daten des Falcon 030 zu präsentieren. Eines kann man vorweg schon sagen: Sollte ATARI den noch unter Vorbehalt angekündigten Preis von ca. 1000,- US-Dollar halten können, steht dem (Home)-Computer-Markt tatsächlich eine Revolution ähnlich der von 1985 bevor, als ATARI den ST vorstellte.

Endlich „details“

Kommen wir jetzt zu den Spezifikationen im einzelnen: der Tabelle 1 können Sie die Leistungsdaten in Kurzform entnehmen. Der Falcon 030 ist (natürlich) mit einem Motorola-68030-Prozessor ausgestattet. Dieser wird mit 16 MHz getaktet, also halb so schnell wie die CPU des TT. Allerdings beträgt auch der Bustakt 16 MHz, so daß mit dieser Geschwindigkeit auf den kompletten Speicher zugegriffen werden kann. Aufwendige Cache-Verfahren, wie sie zur Zeit bei vielen ST-Beschleuniger-Boards (und auch im Mega STE) eingesetzt werden, entfallen beim Falcon 030. Unterstützung erhält die CPU durch einen optionalen, ebenfalls mit 16 MHz getakteten 68882-Coprozessor, für den bereits ein entsprechender (PLCC-)Sockel auf dem Board vorhanden ist. Auf einem internen Erweiterungs-Board lassen sich bis zu 14 Megabyte RAM unterbringen. Auch einen Blitter hat ATARI dem Falcon 030 spendiert. Er ist eine Weiterentwicklung des Mega-ST-Blitters, die nun mit 16 MHz statt 8 MHz läuft. Diese trägt dadurch einen wesentlichen Teil zum Performance-Gewinn in Sachen Grafikausgabe bei. Daß so ein Spezial-Chip auch dringend notwendig ist, merkt man, wenn man sich die Reihe neuer Grafikauflösungen (siehe Tabelle 2) ansieht, die der Falcon 030 nun

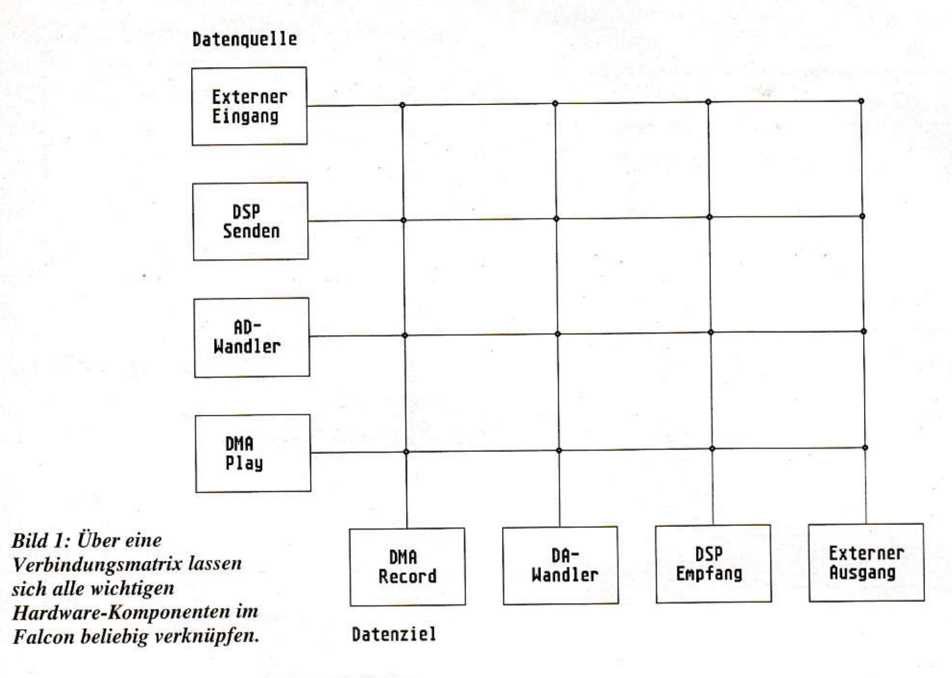


Bild 1: Über eine Verbindungsmatrix lassen sich alle wichtigen Hardware-Komponenten im Falcon beliebig verknüpfen.

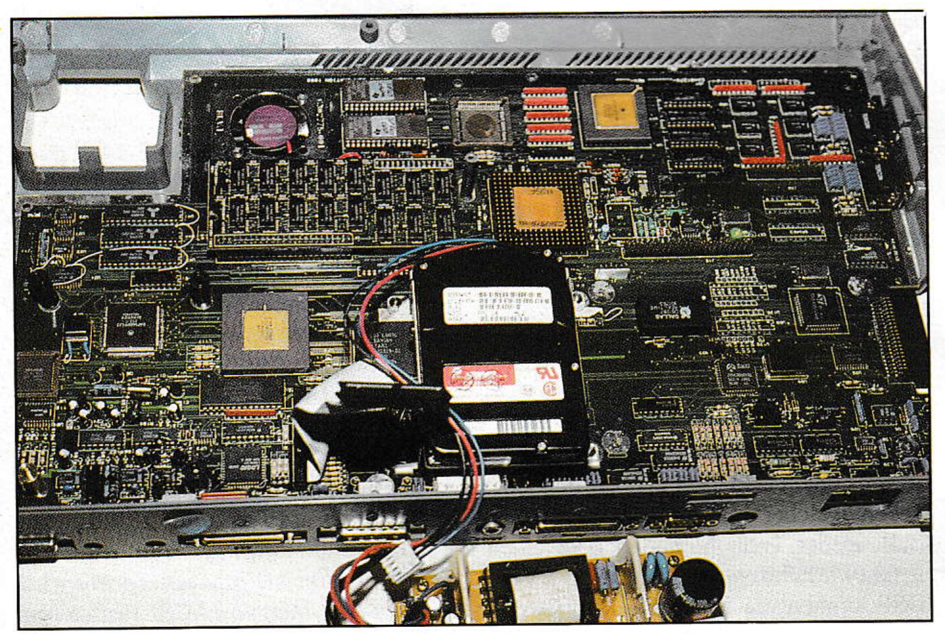


Bild 2: Das Innenleben des Prototypen-Falcon 030 - deutlich erkennt man die 2.5"-AT-Bus-Festplatte.

bietet. Neben den vom ST/STE her bekannten Standardauflösungen findet auch ein (quasi) True-Color-Modus mit 32768 Farben Verwendung; hier müssen 15 Bitplanes (ein Bit wird als Overlay für Genlock-Anwendungen benutzt) verwaltet werden, was einen enormen Datendurchsatz bedeutet.

Programmierbarer Video-Chip

Der Video-Chip, der im Falcon 030 seinen Dienst tut, ist ein von Grund auf neu entwickelter Custom-Chip, der frei programmierbar gestaltet wurde. Dadurch lassen sich über die Standardauflösungen hinaus auch Overscan-Modi verwirklichen, die jeden angeschlossenen Monitor bis zu dessen Leistungsgrenzen ausnutzen können. ATARI nannte hierzu Auflösungen bis 768x576 Pixel (im Interlace-Modus). Durch diese Flexibilität lassen sich verschiedenste Monitortypen am Falcon verwenden. Vom Fernseher (ein HF-Ausgang ist serienmäßig) über RGB-Monitore (SC1224, SC1435, SCART-Fernseher usw.) bis hin zu qualitativ hochwertigen VGA- und Multiscan-Monitoren reicht die Palette der anschließbaren Bildschirme. Selbst der gute alte monochrome Monitor SM-124 verrichtet weiterhin seinen Dienst an ATARIs jüngster Entwicklung. Damit stehen dem Benutzer auch weiterhin alle bekannten Schwarzweißanwendungen zur Verfügung. Zudem kann auch in Monochrom der Overscan-Modus genutzt werden. Es ist natürlich klar, daß bei solchen Möglichkeiten auch professionelle Videoanwendungen wie Genlock usw. ohne zusätzlichen Aufwand zu realisieren sind. In allen Grafikmodi wird Genlock-Synchronisation unterstützt. Daneben hat ATARI aus Kompatibilitätsgründen am horizontalen Hardwarescrolling der STE-Computer festgehalten.

Profi-Sound

Die Entwickler des Falcon 030 haben sich auch bei den Sound-Fähigkeiten Besonderes überlegt. Zum ersten Mal befindet sich ein 16-Bit-Stereo-D/A- und A/D-Wandler serienmäßig in einem Computer. Dazu wurde ein sogenannter „CODEC" (COdierer/DECodierer) eingesetzt, der die Wandlung in beiden Richtungen durchführt. Mit diesen Eigenschaften ist es möglich, den Falcon 030 direkt in eine digitale Stereo-Bandmaschine von CD-Qualität (16 Bit, bis 50 KHz-Sampling-Rate) zu verwandeln. Ein- und Ausgänge für die Audiosignale (Mikrofon/Kopfhörer) sind an der Rückseite in Form von Stereo-Miniklinkenbuchsen herausgeführt. Dadurch läßt sich der Computer wie ein Tonbandgerät in die heimische Stereoanlage integrieren. Die Fähigkeit, gesamplete Sound-Daten direkt auf der Festplatte abzulegen (direct to harddisk-sampling) sorgt dafür, daß die Aufnahmekapazität nur durch die Größe der Festplatte begrenzt ist.

Daneben ist natürlich auch in diesem Bereich auf Kompatibilität zu den Vorgängermodellen Wert gelegt worden. Sowohl der 8-Bit-DMA-Sound als auch der gute alte 3-Kanal-Yamaha-Sound-Chip sind im neuen Falcon 030 wiederzufinden. Letzterer kann sogar als digitaler Wellenformlieferant für den DSP dienen.

Um auch bei Monitoren ohne Lautsprecher (bei vielen VGA- und Multiscan-Geräten üblich) nicht gänzlich auf Sound verzichten zu müssen, hat der Falcon 030 (wie auch schon der TT) einen im Gehäuse untergebrachten Lautsprecher bekommen.

Das Wunderding - der DSP

Viel wurde schon über dieses Wunderding der Technik spekuliert und geschrieben. Nun ist es amtlich: im Falcon 030 kommt ein digitaler Signalprozessor (DSP) vom Typ Motorola DSP-56001 zum Einsatz. Dies allein ist schon eine Sensation, da dieser Chip üblicherweise nur bei Spezialanwendungen der Datenfernkommunikation oder in der Audiotechnik Anwendung

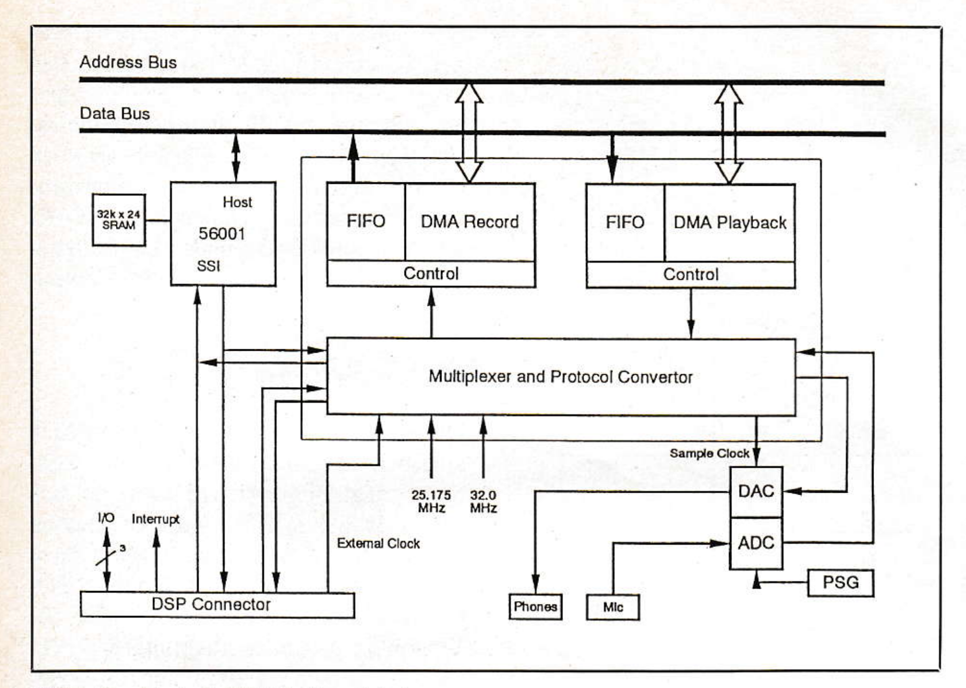


Bild 3: Das Blockschaltbild des Sound-Subsytems

findet. Daß ein DSP aber durchaus seine Daseinsberechtigung in einem Homecomputer haben kann, wird sich zeigen, wenn Sie sich einmal die folgenden Anwendungsmöglichkeiten vor Augen führen.

- Vorstellbar wäre ein Voice-Mail-System, bei dem (ähnlich einer Mailbox) Nachrichten ausgetauscht werden können, die als Sprache und nicht als Text gespeichert werden.
- Natürlich kann man sich damit auch einen Anrufbeantworter (sogar fernsteuerbar) verwirklichen.
- Sprachsynthese sowie Sprachanalyse (Spracherkennung)
- Durch die enorme Geschwindigkeit des DSP, die auch bei der Grafikberechnung (Stichwort: 3D-Vektorgrafik) benutzt werden kann, sind Virtual-Reality-Systeme, die eine künstliche Realität in Echtzeit aufbauen können, denkbar.
- Bild- und Sound-Datenkompression in einem Bruchteil der Zeit, die die momentanen Computer dafür benötigen
- Beeinflussung bzw. Modifikation eingespeister Audio-Signale in Echtzeit, Audio-Effektgeräte wie Echo, Hall, Distortion, Harmonizing etc. lassen sich ohne zusätzliche Hardware erstellen.
- Digitales Mehrspurtonbandgerät. Damit lassen sich nicht nur MIDI-, sondern alle Tonsignale (also auch Sprache, Gesang usw.) wie im Tonstudio bearbeiten und schneiden.
- High-Speed-FAX-Modem ...

... die Liste läßt sich beliebig fortsetzen, der Phantasie sind dabei kaum Grenzen gesetzt. Der Signalprozessor wird mit satten 32 MHz getaktet und hat 32 KB-Worte (ein Wort besteht beim DSP aus drei Bytes, also insgesamt 96 KB) eigenes statisches RAM mit 25 ns Zugriffszeit (0-Wait States). Damit erreicht der DSP eine Rechenleistung von 16 MIPS (Millionen Instruktionen pro Sekunde). Eingebunden ist der Prozessor in eine Verbindungsmatrix (Bild 1) aus DSP, DMA und dem Audio-Subsystem (AD/DA-Wandlung) CODEC. Durch diese Matrix lassen sich beliebige Verbindungen zwischen allen Komponenten herstellen. Es läßt sich z.B. über den 16-Bit-Stereo-AD-Wandler ein Audiosignal einspeisen und auf den DSP geben, welcher es in Echtzeit modifiziert (z.B. in der Tonhöhe ändert oder einen Effekt wie Echo oder Hall hinzufügt). Vom DSP können die Daten direkt über den DMA-Zugfiff auf die Festplatte oder über den 16-Bit-DA-Wandler wieder auf den Audio-Ausgang gelegt werden. Dies alles läßt sich dank der extremen Geschwindigkeit des Systems in Echtzeit, also verzögerungsfrei, ausführen.

Bei diesem Hochgeschwindigkeitsdatenaustausch muß natürlich auch die Datensicherheit gewährleistet sein. Dazu existieren verschiedene Hardware-Handshake-Mechanismen, die Fehler in der Übertragung von einer Komponente zur anderen wirkungsvoll verhindern.

8-Kanal-CD-Sound

Man merkt deutlich, daß ATARI bei der Entwicklung des Falcon 030 besonderen Wert auf das Sound-Subsystem gelegt hat. Nicht nur für den DSP wurde extra eine 26polige Schnittstelle herausgeführt. Dort liegen neben den DSP-Steuersignalen auch jeweils acht digitale Ein- und Ausgänge in einer Auflösung von 16 Bit für das Audio-Subsystem an. Zwar kann man hier nicht direkt analoge Singale einspeisen, aber der dafür notwendige zusätzliche Hardware-Aufwand hält sich in Grenzen, da die seriellen Kanäle dem Industriestandard entsprechen. Im Klartext heißt dies, daß mit entsprechenden ADU-, DAU-Baugruppen eine 2-mal-8kanalige digitale Aufnahmemaschine zu verwirklichen ist. 2 mal 8 Kanäle deswegen, weil die Audio-Subgruppe in der Lage ist, auf 8 Kanälen aufzunehmen und gleichzeitig auf weiteren 8 wiederzugeben. Damit wäre der Falcon, bestückt mit einer hochwertigen DAU-, ADU-Einheit und natürlich einer entsprechenden Festplatte in professionellen Tonstudios (und natürlich auch zu Hause im Heimstudio) als digitale Achtspurtonbandmaschine einsetzbar.

Interne Harddisk

Thema Massenspeicher: ATARI gibt an, daß der Falcon 030 optional mit einer internen 65-Megabyte-2.5"-Festplatte verkauft wird. Dazu ist ein IDE-BUS (auch AT-Bus genannt) vorhanden, an den Standard-AT-Bus-Festplatten angeschlossen werden können. Nach außen wurde ein Standard-SCSI-II-Bus gelegt, der den Anschluß weiterer Festplatten oder anderer SCSI-Geräte (z.B. Laserdrucker/Scanner usw.) ermöglicht. Der vom ST bekannte DMA-Port (ACSI) wurde nicht mehr implementiert. Damit fällt leider ein Stück Hardware-Kompatibilität zum ST weg. Die Megafile-Festplatten lassen sich nicht direkt am Falcon 030 verwenden. Man kann allerdings davon ausgehen, daß es mit Sicherheit Drittanbieter geben wird, die den Anschluß von ACSI-Platten über einen Host-Adapter am SCSI-II-Port ermöglichen.

Intern existiert natürlich auch ein Diskettenlaufwerk. Der Falcon 030 ist serienmäßig mit einem 1.44-MB-HD-Laufwerk ausgestattet, was die Vermutung nahelegt, daß in ihm der AJAX-Chip (ATARI-Nachbau des Floppycontrollers WD-1772 von Western-Digital) seine Arbeit verrichtet. Dieser Chip findet auch schon im TT Verwendung und ist bereits auf ED-Betrieb (also 2.88 MB) vorbereitet. Erst beim zweiten Blick auf die Schnittstellen bemerken wir, daß ein herausgeführter Floppy-Port gänzlich fehlt. Ein zweites Diskettenlaufwerk läßt sich also am Falcon 030 nicht betreiben. Wenn man allerdings davon ausgeht, daß ein Diskettenlaufwerk mittlerweile fast nur noch zum Kopieren der Programme auf die Festplatte benutzt wird, fällt dieser Umstand nicht weiter ins Gewicht.


Zuerst spielte er nach Noten
dann seine eigenen Melodien
und jetzt sahnt er bei der GEMA ab.
Daß ATARI ST Computer die Nr.1 im Musikbereich sind, ist unbestritten. Das MIDI-Interface gehört nun einmal zur serienmäßigen Ausstattung dieses auch in anderen Bereichen erfolgreichen Computers. Es gibt noch eine ganze Reihe weiterer guter Gründe. Da ist die beispielhafte Monochromdarstellung, die Noten exakt lesbar macht. Die Vielzahl an hervorragenden Programmen, denen nur Ihre musikalische Kreativität die Grenzen setzt. So steuern Sie mit Hilfe der gängigen Multitasking-Betriebssysteme gleichzeitig z.B. eine Mischpultautomation, lassen den Sequenzer laufen und verwalten zudem Ihre Sample-Sounds. Und über die Druckerschnittstelle erhalten Sie als Noten, was Sie somit Schwarz auf Weiß als Ihr eigenes Werk vielleicht zu hohen Ehren kommen läßt.
Wann erweitern Sie Ihr musikalisches Repertoire mit einem ATARI STE Computer? Sprechen Sie mit Ihrem ATARI MIDI/Musik-Fachhändler darüber. Das ist:
ATARI
und Musik
ATARI MEGA STE
1 oder 2 oder 4 MB RAM
integrierte Floppy und Festplatte
serienmäßig mit flimmerfreiem
s/w Monitor 71 Hz
ATARI SLM 605
Laserdrucker für
gestochen scharfe s/w Ausdrucke
Für ATARI STE Computer
gibt es Sequenzer und
Editoren von C-Lab,
Soft Arts und Steinberg,
Lernsoftware von Schott
und Harddiskrecording
von Hybrid Arts,
um nur einige zu nennen.
ATARI Computer GmbH
Postfach 12 13
6096 Raunheim
ATARI
...wir machen Spitzentechnologie preiswert
ATARI und Musik • ATARI und Textverarbeitung • ATARI und Datenbanken • ATARI und Spaß mit Grips •
ATARI und DeskTop Publishing • ATARI und Büro • ATARI und Studium • ATARI und Wissenschaft • AT

Das Innenleben

Auf der Pressekonferenz wurde noch ein Prototyp des Falcon 030 gezeigt. Dennoch erlaubte man uns, einen Blick in das Innere des Gerätes zu werfen (Bild 2). Interessant ist der interne Erweiterungs-Port, der aus zwei diagonal versetzt angeordneten Pfostenfeldleisten besteht. An diesem Port liegen alle wichtigen Hardware-Signale des Rechners an. Erweiterungen wie MS-DOS-Emulatoren brauchen also nicht, wie bislang bei den STs, in abenteuerlichen Konstruktionen auf den Prozessor gelötet zu werden. Deutlich erkennt man das 2,5"-Festplattenlaufwerk. Natürlich entsteht bei soviel Rechen-Power ein gewisser Grad an Abwärme; Besitzer von 1040ST/STEs können ein Lied davon singen. Schließlich befindet sich wie beim 1040ST/STE das Netzteil gleich mit im Gehäuse. Um Hitzeschäden vorzubeugen, hat man daher einen sehr flachen Lüfter direkt auf der Platine angebracht. Dieser sorgt für ausreichende Kühlung.

Die Software-Unterstützung

Das schon zur CeBIT '92 vorgestellte MultiTOS wird erstmalig im Falcon 030 serienmäßig zum Einsatz kommen. Dieses TOS ist weitestgehend kompatibel zu früheren Versionen, ermöglicht jedoch Multitasking. Wir haben schon an anderer Stelle ausführlich darüber berichtet. Eine echte Neuheit ist allerdings, daß die speziellen Hardware-Fähigkeiten des Falcon 030 auch vom TOS unterstützt werden. Es wird Betriebssystemfunktionen geben, mit deren Hilfe man DSP-Programme installieren, ausführen bzw. deren Vorhandensein abfragen kann. Somit lassen sich beispielsweise bestimmte DSP-Anwendungen quasi als Unterroutinen aus jedem Programm heraus aufrufen. Motorola bietet schon eine ganze Reihe von DSP-Anwendungen als Public-Domain an, so daß hier auf ein erhebliches Potential zurückgegriffen werden kann. Auch wird es Funktionen zur Programmierung des neuen Video-Chips geben. Der Entwickler braucht sich somit nicht mehr mit speziellen Hardware-Eigenschaften dieses Chips auseinanderzusetzen.

Das Design

Zur CeBIT '92 hieß es noch, daß der Falcon 030, um sich optisch deutlich vom 1040ST/STE abzuheben, ein dunkelgraues Gehäuse bekommen würde. Von diesem Standpunkt ist ATARI mittlerweile abgewichen. Damit auch die alte ST-Peripherie (z.B. Monitor) optisch noch zum Falcon 030 paßt, hat man sich auf einen Kompromiß geeinigt. Das Gehäuse bleibt im gewohnten hellen ATARI-grau, lediglich die Tastenkappen werden in Dunkelgrau mit weißer Aufschrift gehalten sein.

Falcon 030 - ein neuer Stern am Computer-Himmel?

Selbstverständlich können wir hier nicht alle Möglichkeiten dieses neuen ATARI-Computers erläutern. Sie sind so komplex, daß es selbst uns schwerfällt, sie zu überschauen. Vieles wird sich erst im Laufe der Zeit ergeben, und die Anwenderschaft wird sicherlich mit der einen oder anderen Sensation aus den Reihen der Programmierer und Freaks überrascht werden. Eines ist jedoch gewiß: noch nie wurde ein so leistungsfähiges Hardware-Konzept in einem Homecomputer für einen Endpreis von unter 2000,- DM verwirklicht. Bei entsprechender Unterstützung der Software-Häuser stehen die Chancen für eine erfolgreiche Markteinführung des Falcon 030 mehr als gut. Der zur Zeit etwas müde Markt braucht wieder einen Computer, der Begeisterung auslösen kann. Wir glauben: mit dem Falcon 030 hat ATARI eine solche Maschine entwickelt. Auch wenn die Freaks und Profianwender zunächst vor einem Consolen-Computer zurückschrecken werden; der Falcon 030 ist der Beginn einer ganzen Reihe von neuen ATARI-Computern. Ein Gerät mit abgesetzter Tastatur und Erweiterungs-Slots wird, laut ATARI, in Kürze folgen. Bleibt nur zu hoffen, daß man aus den Fehlern der Vergangenheit gelernt hat und der Erfolg nicht wieder an Lieferschwierigkeiten erstickt.

CM

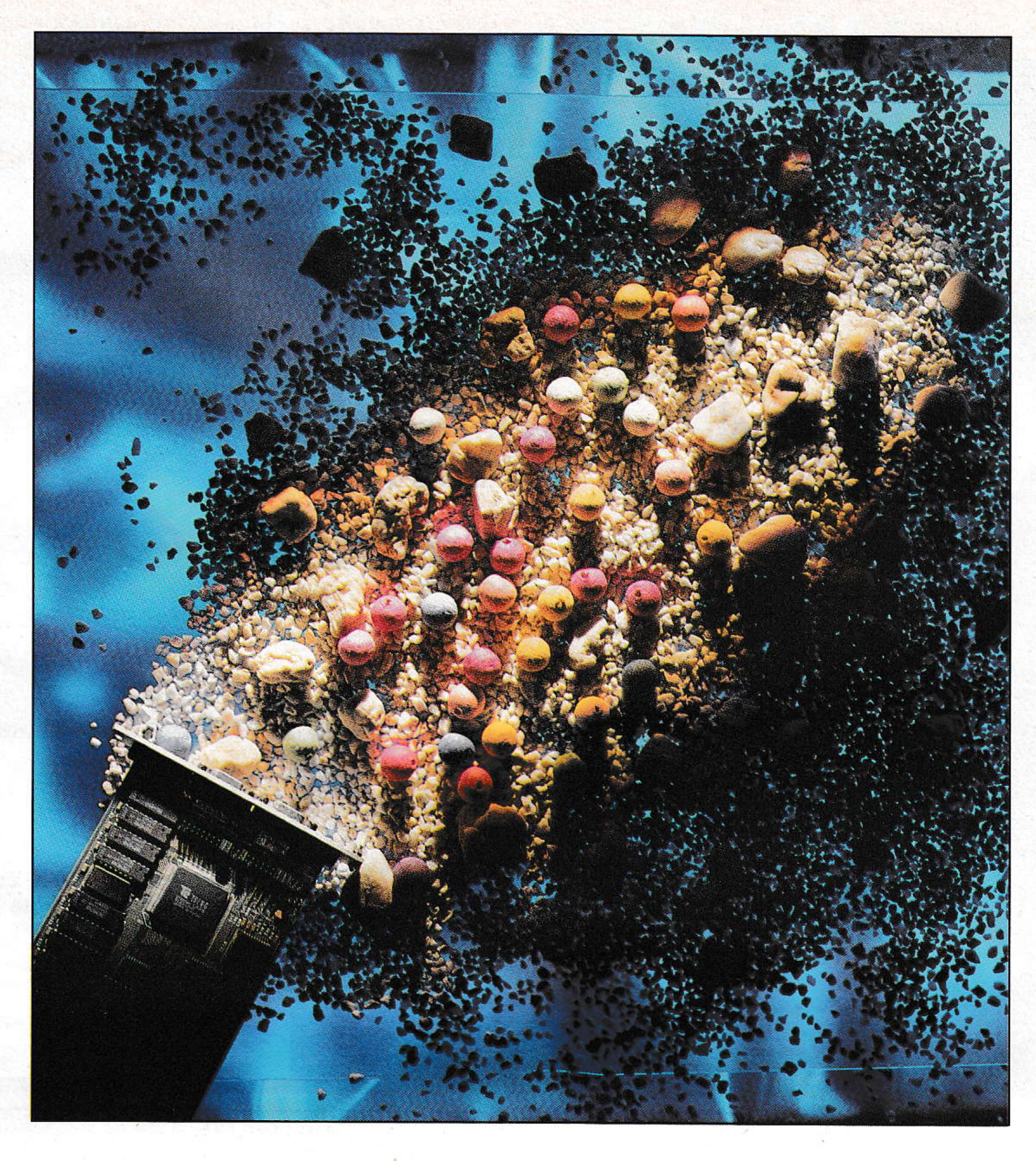
Prozessor:	Motorola 68030 16 MHz
Coprozessor:	Motorola 68882 16 MHz (optional)
RAM:	4 MB (bis 14 MB intern aufrüstbar)
ROM:	512 KB
Floppy:	1,44MB 3,5" HD-Laufwerk intern
Festplatte:	65 MB 2.5" IDE-Bus intern (optional)
DSP:	Motorola DSP 56001 32 MHz plus 32 Kworte (=96KB) statisches RAM
Grafik:	siehe Tabelle 2
Sound:	16 Bit Stereo Digital-Sound
	8 Bit Stereo DMA-Sound (STE/TT kompatibel)
	3 Kanal Yamaha-Soundchip (ST-kompatibel)
	2 mal 8 16 Bit digitale Sound-Kanäle
TOS:	MultiTOS
Schnittstellen:	
extern:	
1 x seriell	(9pol. Sub-D male)
1 x parallel	(25pol. Sub-D female)
1 x SCSI II	(50pol.)
1 x Mikrofon	(Stereo-Miniklinke)
1 x Kopfhörer	(Stereo-Miniklinke)
1 x Monitor/Video/Genlock	(19pol. Sub-D male)
1 x DSP/Digital Audio Port	(26pol. Sub-D female)
1 x LAN-Port	(9pol. Mini-DIN)
1 x TV-Tuner (HF)	(RCA female)
1 x MIDI in	(5pol. DIN female)
1 x MIDI out	(5pol. DIN female)
1 x ROM-Port	(40pol. ATARI)
2 x erweitere Joysticks	(15pol. Sub-D male)
2 x Maus/Joystick	(9pol. Sub-D male)
intern:	
DRAM-Expansion-Bus	(30Pin + 50Pin Pfostenfeld)
Expansion-Bus	(30Pin + 50Pin Pfostenfeld)
IDE-Bus	

Tabelle 1: Kurzübersicht Falcon 030

Modus	Auflösung	Bitplanes	Farben	Palette
ST-Low	320x200	4	16	4096
ST-Med	640x200	2	4	4096
ST-High	640x400	1	2	4096(duochrom)
True-Color	320x200	15	32768	keine (1 Bit für Overlay-Modus)
Falcon	640x480	8	256	262144
VGA/Video	x: 320 oder 640 y: 200 oder 400 Overscan möglich	1,4,8	2,16,256	262144

Tabelle 2: Grafik-Auflösungen des Falcon 030

Bunter Pixel-Strom



Atari hat die Rechnermodelle STE und TT nicht gerade mit üppigen Grafikmodi ausgestattet: Vor allem bei Farben und Graustufen zeigen sie sich deutlich unterbelichtet, wie ein Blick nicht nur in die DOS-Welt zeigt. Gleichwohl fand der VME-Bus als Erweiterungs-Port in beiden Rechnern Platz. Naheliegend, daß es mittlerweile ein reichliches Angebot an Grafikkarten für diesen Erweiterungs-Slot gibt.

Der immer stärkere Einsatz von Bildverarbeitung und DTP macht die Ausgabe der Werke auf dem Monitor in 256 und mehr Farben oder Graustufen notwendig. Wir haben uns daher Karten herausgesucht, die dem STE/TT in dieser Hinsicht Zuwachs bescheren. Das sind die sechs Modelle COCO (Matrix), Crazy Dots (TKR), Imagine (Wittich), MEGA-Vision (Sang/Trade iT), Spektrum (Wilhelm) und TC1208 (Matrix). Die immer wieder angekündigte Chroma (Omega) stand uns zum Zeitpunkt des Tests nicht zur Verfügung. Bei der Crazy Dots und der Imagine handelt es sich um an den VME-Bus angepaßte VGA-Karten, bei den anderen Modellen sind sowohl Layout der Hardware als auch die Software Eigenentwicklungen der jeweiligen Hersteller.

Konzept

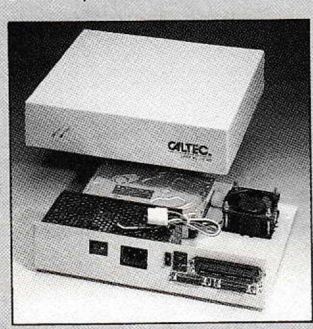
Genaugenommen handelt es sich bei den Karten um Grafiksubsysteme. Denn es wird dem Rechner nichts von seiner ursprünglichen Fähigkeit genommen. Ein Monitor, der am Rechner bereits angeschlossen ist, kann dort bleiben und weiterhin genutzt werden. Im VME-Slot steckt die Karte, die auf Wunsch weitere Grafikmodi zur Verfügung stellt. An sie ist ein (ggf. zweiter) Monitor angeschlossen.

Wie stellt nun das Betriebssystem fest, daß die Ausgabe auf einen anderen als den normalen Monitor zu erfolgen hat, besser gesagt: daß der Bildschirmspeicher an anderer Stelle liegt sowie anderen Aufbau und Größe besitzt? Zwischen Karten und Betriebssystem steht immer ein Treiber, der für die nötige Verständigung sorgt. Er dient dazu, die Ausgaben der verschiedenen Anwendungen auf den rechten Weg zu leiten.

Für diese Treiber gibt es unterschiedliche Konzepte. Bei der Crazy Dots beispielsweise werden lediglich drei Treiber-Dateien in den GEMSYS-Ordner kopiert und in der ASSIGN.SYS-Datei angemeldet. Ein kurzes Steuerprogramm im Auto-Ordner läßt die Wahl der Farben und der Auflösungsgrößen zu. Bei der COCO und der TC1208 von Matrix werden ein eigenes BIOS, eine angepaßte Version des AMCGDOS und ein entsprechendes Steuerprogramm im Auto-Ordner benutzt. Fast alle Konzepte brauchen also ein installiertes GDOS; teils nehmen sie mit dem etwas schnelleren NVDI-GDOS vorlieb.

Anschlußfertige SCSI-Festplatten für Atari TT und Mega STE

▶ Vom Design passend zur Haupteinheit des TT bzw Mega STE.
▶ Bei Anschluß an TT können unsere Geräte optional ohne SCSI-Hostadapter betrieben werden, bei Kauf einer anschlußfertigen Festplatte ohne SCSI-Hostadapter reduziert sich der Kaufpreis um DM 150,–
▶ mit gesteuertem, sehr leisem Lüfter
▶ weitere technische Daten wie unsere SCSI-Festplatten für die Atari ST-Serie.



Seagate ST157N-1, 49 MB, 28 ms	798,-
Quantum LPS52S, 52 MB, 17 ms	898,-
Seagate ST1096N, 81 MB, 24 ms	1.098,-
Quantum LPS105S, 105 MB, 17 ms	1.198,-
Seagate ST1239N, 200 MB, 15 ms	1.798,-
Imprimis ST1480N, 425 MB, 14 ms	2.998,-
Imprimis ST4766N, 676 MB, 14 ms	4.998,-
Syquest SQ555, 44 MB, 25 ms	1.398,-

SCSI-Kits

Unsere SCSI-Einbaukits bestehen aus Festplattenlaufwerk, SCSI-Hostadapter, DMA-Kabel, SCSI-Kabel, Installationssoftware und Handbuch.

Seagate ST157N-1, 49 MB, 28 ms, Kit	598,-
Quantum LPS52S, 52 MB, 17 ms, Kit	698,-
Seagate ST1096N, 81 MB, 24 ms, Kit	898,-
Quantum LPS105S, 105 MB, 17 ms, Kit	998,-
Seagate ST1239N, 200 MB, 15 ms, Kit	1.598,-
Syquest SQ555, 44 MB, 25 ms, Kit	1.098,-

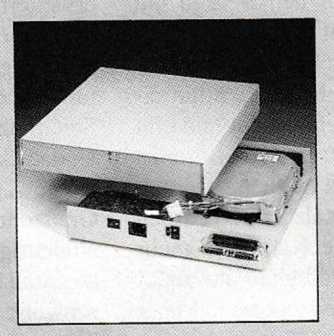
Nachfolgend diverse für SCSI-Festplattenlösungen benötigte Einzelteile:

SCSI-Komplettkit bestehend aus Gehäuse, Netzteil, Hostadapter, Kabel, Software, Handbuch, Montagematerial	398,-
SCSI-Hostadapter incl. Software, Kabel	198,-
Syquest-Cartridge SQ400	198,-
Gehäuse für HDD	99,-
Netzteil 50 Watt	99,-
DMA-Kabel	39,-
SCSI-Kabel	39,-

Hoch-stimmung über Preistief

Anschlußfertige SCSI-Festplatten für die Atari ST-Serie

▶ Datentransferraten > 600 KByte/s, mit Imprimislaufwerken bis zu 1350 KByte/s erzielbar, Zugriffszeiten bis zu 14 ms.
▶ Spitzensoftware: 255 Partitionen installierbar, Softwareschreibschutz, jede Partition autobootfähig, Interleave 1 : 1 einstellbar, Cache, Backup, Optimizer in der Software enthalten.
▶ 100% Atari-kompatibel, Fremdbetriebssysteme (PC-Speed, AT-Speed, PC-Ditto, Spectre, Aladin, Minix, OS-9, RTOS) sind voll lauffähig.
▶ Sehr leise, 3.5"-Festplatten ohne Lüfter, 5.25"-Festplatten mit gesteuertem Lüfter.
▶ Durchgeschleifter gepufferter DMA-Bus, Autoparkfunktion hardwaremäßig, DMA-Nr. von außen einstellbar.
▶ Herausgeführter SCSI-Bus, durch abschaltbaren Hostadapter optimale Datenübertragung auch an Apple McIntosh und PCs.
▶ Zweite SCSI-Festplatte im Gehäuse nachrüstbar, SCSI-Hostadapter und Gehäuse für interne zweite Festplatte vorbereitet.



Seagate ST157N-1, 49 MB, 28 ms	798,-
Quantum LPS52S, 52 MB, 17 ms	898,-
Seagate ST1096N, 81 MB, 24 ms	1.098,-
Quantum LPS105S, 105 MB, 17 ms	1.198,-
Seagate ST1239N, 200 MB, 15 ms	1.798,-
Imprimis ST1480N, 425 MB, 14 ms	2.998,-
Imprimis ST4766N, 676 MB, 14 ms	4.998,-
Syquest SQ555, 44 MB, 25 ms	1.298,-



SENSATIONELL.

Die Workstation für zu Hause

Atari TT mit 32 MHz
Prozessor MC 68030 +
Coprozessor 68882
VME-Steckplatz,
6 Grafikmodi,
2MB RAM, Maus **1.999,-**

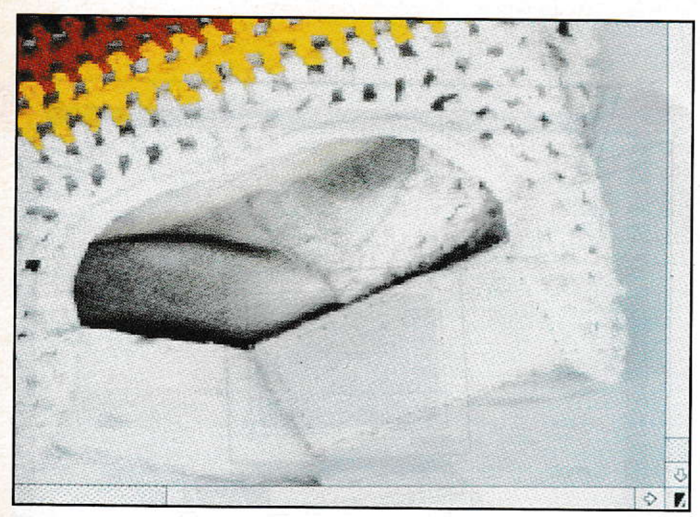
COMPUTERSYSTEME
Schlichting
...das innovative Systemhaus
Computersysteme Playsoft - Studio Schlichting
Computer - Software - Versand GmbH & Co KG

Geschäftsstellen
1000 Berlin 61 • Katzbachstraße 8 • 1000 Berlin 20 • Mönchstraße 8
1560 Potsdam • Charlottenstraße 56

Zentrale
Postanschrift • Verwaltung • Versand
Katzbachstraße 8 • 1000 Berlin 61

Zentrale Telefonnummer für alle unsere Geschäftsstellen:
030 / 786 10 96 Telefax: **030 / 786 19 04**

AGM



True-Color-Darstellung mit 16,7 Millionen Farben. Optimale Darstellung mit maximaler Schärfe.



Die Darstellung in 32.768 Farben zeigt nur geringe Verfälschungen.

Darüber hinaus gibt es während des Betriebs der Karten einige Einstellungen zu tätigen. Das sind natürlich die Farbpaletten, für deren Kontrolle allen Produkten ein entsprechendes Accessory beiliegt. Aber auch Bildlage und evtl. Dunkelschaltungen wollen gesteuert werden. Dafür stehen ebenfalls ACCs zur Verfügung. Leider sind diese nur bei der Spektrum platzsparend als CPX-Module ausgebildet.

Die Treiber sind im allgemeinen VDI-Treiber, denn das VDI stellt die unterste Schicht des grafischen Teils des Betriebssystems dar. (Darüber liegt das AES, das sich der vom VDI zur Verfügung gestellten Funktionen bedient.) Das VDI ermöglicht die Ansteuerung grafikorientierter Ausgabegeräte. Da jede ‚ordentliche' Grafikausgabe auf dem ST über das VDI geht, muß der Treiber die VDI-Funktionen auf die Grafikkarte abbilden, deren Hardware ja nur er kennt. Mehr steht darüber in [1].

Lücken

Wie ein roter Faden zieht sich die VDI- und Treiberproblematik durch diesen Test. An dieser Stelle liegen die Ursachen für (fast) alle Probleme, die man mit einer Grafikkarte so haben kann; z.B. daß bestimmte Anwendungen nicht oder nicht richtig laufen. Dann bedienen sie sich vermutlich nicht nur der VDI-Funktionen, sondern schreiben direkt in den Bildschirmspeicher. Diese Art Programm scheidet für den Betrieb mit einer der hier getesteten Karten sofort aus.

Häufig verrichtet ein Programm seinen Dienst, erkennt aber z.B. die Echtfarbfähigkeit des Grafiksystems nicht und gibt die Bilder in 256 statt in 16 Millionen Farben aus. Der Grund dafür liegt darin, daß im VDI nicht mehr als 32.768 Farben verwaltet werden. Jede Karte, die mehr beherrscht, muß ihre Fähigkeit auf anderem Wege kundtun. Meist tut sie das über einen Cookie-Eintrag. Doch die Applikation muß darauf vorbereitet sein. D.h. daß jedes Programm, das ein Echtfarbbild ausgeben will, sämtliche Cookie-Einträge kennen muß, also auf jede Karte angepaßt werden muß. Nur so ist eine hundertprozentige Sicherheit gewährleistet. Dieses schwere Los haben uns die Väter des GEM aufgebürdet, indem sie es nicht zeitgemäß erweitert haben.



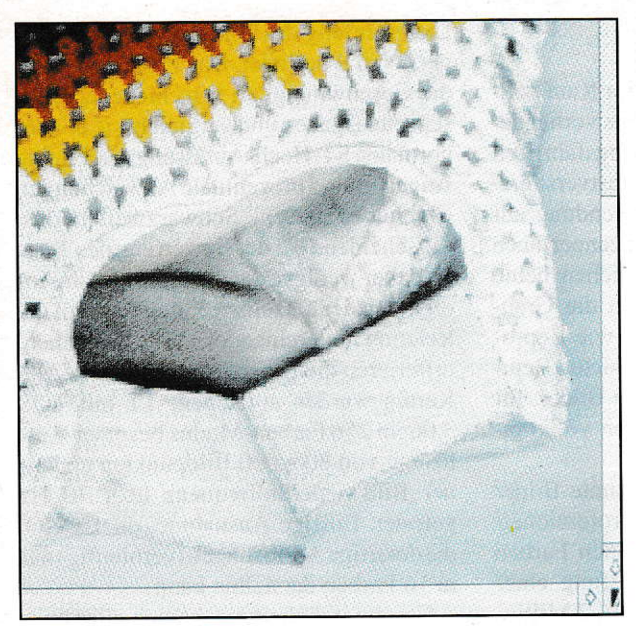
Dieser in 16,7 Millionen Farben eingescannte Handschuh wird in den verschiedenen Darstellungsmodi der Grafikkarten auf den Schirm gebracht.

Nun besteht kein Grund zur Panik, denn die gängige Software zur Bildverarbeitung (Cranach, Calamus, ReproStudio, SciGraph) wird angepaßt, und über ein Update ist die Lauffähigkeit auf allen Karten gewährleistet. Allerdings besteht eine Grauzone von Problemen, die nicht immer eindeutig sind. Da wird z.B. der Texthintergrund beim Texteditor mit einem Male schwarz, woran weder der Kartenhersteller noch der Programmautor schuld sein wollen. Da hilft nur Warten auf ein Update oder der Umstieg auf ein Programm, das GEM-konformer verfaßt ist. Unter den Textverarbeitern ist das mit Sicherheit Script, bei den Texteditoren z.B. Edison.

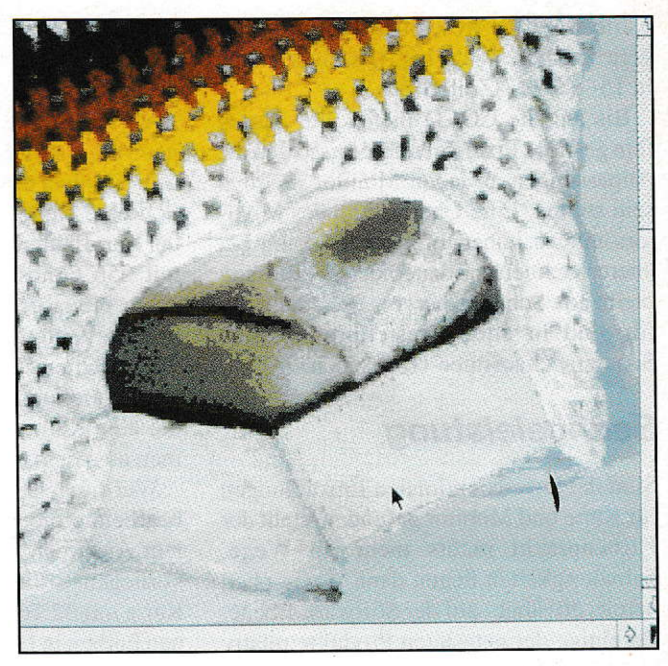
Die wahre Farbenpracht

Die hier getesteten Karten bieten allesamt 256 und mehr Farben. Das heißt, sie können jedem Bildschirmpunkt eine von 256 verschieden Farben zuordnen. Das können auch Grauabstufungen sein, dann spricht man von 256 Graustufen. Dieser Modus empfiehlt sich bei der Bearbeitung gescannter Bildvorlagen, die aus Graustufen-Scannern stammen. Bei 256 Farben benötigt jeder Bildschirmpunkt den Speicherplatz von einem Byte (Wertebereich 0-255).

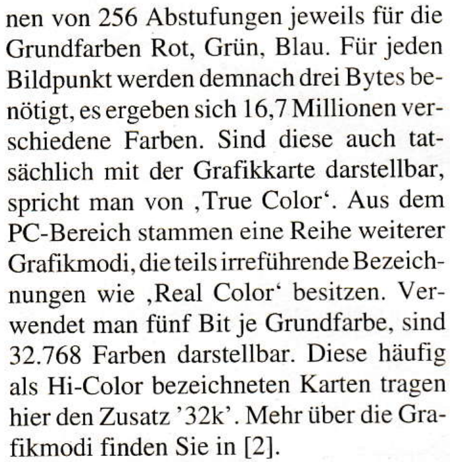
Ein sogenanntes Echtfarbbild trägt auf jedem Bildschirmpunkt die Informatio-



Geditherte Darstellung in 256 Farben. Das Bild wird leicht unscharf.



Darstellung in 256 Farben. Farbverfälschungen sind deutlich erkennbar.



nen von 256 Abstufungen jeweils für die Grundfarben Rot, Grün, Blau. Für jeden Bildpunkt werden demnach drei Bytes benötigt, es ergeben sich 16,7 Millionen verschiedene Farben. Sind diese auch tatsächlich mit der Grafikkarte darstellbar, spricht man von ‚True Color‘. Aus dem PC-Bereich stammen eine Reihe weiterer Grafikmodi, die teils irreführende Bezeichnungen wie ‚Real Color‘ besitzen. Verwendet man fünf Bit je Grundfarbe, sind 32.768 Farben darstellbar. Diese häufig als Hi-Color bezeichneten Karten tragen hier den Zusatz '32k'. Mehr über die Grafikmodi finden Sie in [2].

Stellt man Farbbilder, die 16,7 Millionen verschiedener Farben besitzen, auf einer Karte mit 256 Farben dar, ergibt sich ein Effekt, wie Sie ihn im Bild mit dem Handschuh sehen können. Da es zu einer Reduzierung der Information kommt, werden beieinander liegende Farben des ursprünglichen Bildes einer einzigen im dargestellten Bild zugeordnet. Gerade im Bereich der nuancierten Übergänge macht sich dieser Effekt störend bemerkbar.

Daher bedient sich die Software eines Verfahrens, das ‚Dithering‘ genannt wird. Durch Rastermuster werden aus verschiedenen Farben Zwischentöne gebildet - natürlich lassen diese Raster das Bild unschärfer wirken. Zudem kostet das Verfahren teils erheblich Rechenzeit, die bei jedem Neuaufbau des Bildes anfällt. 32.768 Farben lassen auch ohne Dithering bereits eine sehr getreue Farbdarstellung zu, lediglich in sehr seichten Übergängen treten störende Verfälschungen auf. In der True Color-Darstellung muß das Bild nicht gedithert werden, und keinerlei Paletten sind mehr für die Zuordnung der Farben zuständig. Dadurch ist diese 16,7-Millionen-Farben-Darstellung interessanterweise nicht langsamer als die mit 256 Farben plus Dithering.

Handarbeit

Kehren wir zu unseren Testkandidaten zurück und betrachten kurz den Vorgang der Installation. Nachdem die Karte in den VME-Slot gestöpselt wurde, heißt es, die Treiber-Software richtig zu installieren. Wie bereits erwähnt, ist dies eine unterschiedliche Anzahl von Programmen, Accessories und Parameterdateien, die in den richtigen Pfaden der Boot-Partition stehen müssen. Da kommt DOS-Feeling auf, und nicht immer greifen dem Hilflosen etwaige Installationsprogramme unter die Arme. Und selbst wenn, so installieren sie zumeist selbstherrlich ihre Version der ASSIGN.SYS, anstatt sie nur anzupassen. Lediglich auf der Diskette zur Crazy Dots und bei den Karten von Matrix fand sich ein Boot-Selektor. Der ist auch dringend nötig, will man nicht permanent auf der Karte arbeiten, sondern vielleicht auch noch mit anderen GDOS-Installationen. Da ist besonders XBOOT hilfreich, denn das kopiert ja auch die ASSIGN.SYS um.

Sind dergleichen Hilfen nicht gegeben, muß in mühsamer Handarbeit kopiert werden. Ist die Karte erst in Betrieb, so bieten Einstellprogramme die Möglichkeit, Auflösung und Anzahl der Farben zu verändern. Dabei müssen die Werte der horizontalen und vertikalen Auflösung, der jeweiligen Ränder, Pixel-Frequenzen und Farbenanzahl sowie - falls möglich - die Größe der virtuellen Auflösung bestimmt werden. Näheres zu diesen Werten in [2]. Bis auf die MEGA-Vision bieten alle Treiber die Möglichkeit einer virtuellen Auflösung. Dabei wird dem Rechner ein viel größerer Bildschirm vorgegaukelt, als tatsächlich vorhanden ist. Wird der Rand des tatsächlichen berührt, scrollt der Treiber mittels Hardware-Scrolling weiter, was sehr schnell geht. So spart man sich lästige Redraw-Zeiten bei der Arbeit.

Parameter für verschiedene Geräte werden mitgeliefert, doch oft genug fehlt genau die für den heimischen Monitor, oder man möchte aus der eigenen Röhre das Letzte herauskitzeln. Die Lösungen, zu eigenen Werten zu gelangen, differieren dabei erheblich. Optimal sind die Programme, die eine interaktive Zwei-Schirm-Lösung bieten. Auf dem normalen Monitor wird dabei das Programm-Menü gezeigt, ein Testbild auf dem Monitor, der an die Karte angeschlossen ist, kündet vom Erfolg der Einstellungen. Überzeugen können dabei die Programme zur Imagine und zu den beiden Matrix-Karten.

Vor dem Kauf der Grafikkarte sollten Sie einen Blick ins Handbuch Ihres Farbmonitors werfen: Handelt es sich um einen echten Multiscan, der sich automatisch auf die ihm gelieferten Frequenzen einstellt, oder ist es ein VGA-Schirm, der feste Frequenzen benötigt? (Daß er evtl. ein ‚Multi‘ im Namen trägt, muß nichts bedeuten!) In jedem Fall sollte die Grafikkarte die Bereiche der Horizontal- und Vertikalfrequenz abdecken, die der Monitor synchronisieren kann. Ist das nicht der Fall, verschenken Sie Leistung. Zum Beispiel bietet die MEGA-Vision Pixel-Frequenzen bis zu 110 MHz, während ein Standard-VGA-Monitor wie der Atari PTC 1426 lediglich 30 MHz verkraftet. Diese

Karte wäre für wesentlich leistungsfähigere Monitore (siehe in [2]) geeignet, auch in 16- oder 17-Zoll-Größe.

Bis auf die COCO bieten alle Karten fein gestufte Pixel-Frequenzen, so daß die Anpassungen an jeden Monitor möglich sein sollten. Eine Überlastung des Monitors mit zu hohen Frequenzen kann bei älteren Geräten und dauernder Überlastung bleibende Schäden hervorrufen, doch dafür sind in allen Einstellprogrammen Schutzmechanismen vorgesehen.

Serviceleistung

Sind die jeweils optimalen Einstellungen für Karte und Monitor gefunden, steht der Farbenpracht nichts mehr im Wege. Schwierigkeiten liegen dann - wie oben bereits erwähnt - auf der Software-Ebene. Da einige der getesteten Produkte relativ neu am Markt sind, bedürfen die Treiber hier und da der Nacharbeit. Updates sind häufig vonnöten, wofür die Firmen Sang, Wilhelm und TKR eigene Service-Mailboxen unterhalten. Wer ein Modem besitzt, kann sich dort jederzeit die aktuellen Treiber beschaffen.

Kriterien

Entscheidungsgrundlage beim Kauf sollten, wie gesagt, ein evtl. bereits vorhandener Monitor und seine Leistungsfähigkeit sein. Darüber hinaus spielt selbstverständlich eine Rolle, welche Anwendung auf der Karte laufen soll. Handy-Scanner, auch mit 256 Graustufen, ermöglichen heute einen preiswerten Einstieg in die Graubildbearbeitung. Programme wie Repro-Studio, Cranach oder Charly bieten Retuschiermöglichkeiten für diese Bilder, für die eine Karte mit 256 Farben vollkommen ausreicht.

Wer hingegen farbig gescannte Bilder bearbeiten möchte, die mit professionellem Anspruch in 16,7 Millionen Farben vorliegen, sollte die Anschaffung einer Karte mit 32.768 oder gleich 16,7 Millionen Farben in Erwägung ziehen. Denn die zeit- und qualitätsintensive Dither-Methode verdirbt schnell die Freude. Zu beachten ist allerdings, daß die MEGA-Vision und die Spektrum in ihren True-Color-Modi lediglich eine Auflösung von 640x480 Bildpunkten ermöglichen, da sie nur über 1MByte Speicher verfügen.

Die Testergebnisse, die Sie zu den einzelnen Karten finden, wurden mit dem GEM_TEST.PRG ermittelt, das sich im Lieferumfang des Bildschirmbeschleunigers NVDI befindet. Das Programm mißt die Geschwindigkeit verschiedener Bildschirmausgaben und vergleicht sie mit der normalen ST-Hoch-Auflösung des TT. Die Angaben sind prozentuale Abweichungen gegenüber diesem Schwarzweißmodus. Entscheidend bei der Bewertung des Tests ist dabei weniger die absolute Geschwindigkeit jeder einzelnen Karte, sondern die Relation untereinander. Wie Sie sehen, differieren die Ergebnisse sehr stark. Alle Karten wurden auf einem TT mit TOS 3.06 im 256 Farben-Modus bei einer Auflösung von 800x600 Bildpunkten und einer Bildwiederholfrequenz über 70 Hz getestet. Einzige Ausnahme: die COCO, die derartige Auflösungen (vernünftig) nur in 16 Farben darstellen kann.

Geschwindigkeit macht sich allerdings nicht nur in Meßwerten bemerkbar. Wesentlich entscheidender ist der subjektive Eindruck bei der Arbeit mit dem Rechner. Dafür sind die von GEM_TEST beurteilten Punkte ‚Linien', ‚Text', ‚Rechtecke' und ‚AES-Ausgabe' entscheidend. Eine Karte, die in diesen Punkten gut abschneidet, hinterläßt auch subjektiv einen schnellen Eindruck. Auffällig an den Testergebnissen ist, daß die BIOS-Ausgabe fast immer langsamer ist als die (ebenfalls

Karte	COCO	Crazy Dots 32k	Imagine 32k	MEGA-Vison 300	Spektrum 1TC	TC1208
Hersteller	Matrix, Oppenweiler	TKR, Kiel	Wittich, Abensberg	Sang, Essen/ Trade iT, Roßdorf	Wilhelm, Lünen	Matrix, Oppenweiler
Video-RAM	1 MB	1 MB	1 MB	1 MB	1 MB	2 MB
max. Farben	256	32.768	32.768	16,7 Millionen	16,7 Millionen	16,7 Millionen
Pixelfreq.	16/25/28/50 MHz	14 Frequenzen zwischen 14 und 65 MHz	25/32/40/45/52/ 65/72/75/80/90 MHz	5/7/14/28 MHz sowie 25-110 MHz in 5 MHz-Schritten	25/28/32/36/40/45/ 56/60/65/80/84 MHz	5-132 MHz
Preis (unverb. Empf. laut Hersteller)	1099 DM	1198 DM (256-Farbversion: 998 DM)	998 DM (256-Farbversion: 798 DM) (Upgrade 248 DM)	1498 DM	1398 DM (32.768-Farbversion: 998 DM) (256-Farbversion: 748 DM) (Upgrades möglich)	3990 DM (1-MB-Version: 2490 DM)
Software:						
virt. Auflösung	ja	ja	ja	nein	ja	ja
ACC zur Einst. der Farben und derBildlage	ja	ja	ja	nur Farben	ja, als CPX-Modul	ja
ext. Programm zur Anpassung neuer Monitore	ja	ja	ja	nein, in Vorbereitung	ja	ja
Bewertung:						
Geschwindigkeit	–	+	+	–	0	+
Bildqualität	+	+	0	0	0	+
Software	+	+	+	–	+	+
Ausstattung	–	+	+	–	0	+

schleppende) des normalen TT-Hoch-Modus. Das liegt daran, daß die VDI-Treiber, obwohl sie dafür ja eigentlich nicht zuständig sind, diese Ausgaben abfangen und in Grafik wandeln müssen. Außer der ,Anzeigen'-Funktion des Desktops wird diese Art der Textdarstellung aber ohnehin von keinem ordentlichen Programm benutzt.

Ferner wurde noch ein Test im Monochrombetrieb durchgeführt. Denn wer lediglich mit einem Monitor arbeiten möchte, wird nicht umhinkommen, auf einfarbigen Betrieb zu wechseln. Es gibt eine Reihe Programme, vor allem Textverarbeitungen, die zwar in jeglicher Auflösung, aber nicht in Farbe laufen. Da heißt es vor jedem Start, neu zu booten. Lediglich die Imagine verfügt über ein Programm, das in der Lage ist, abhängig von der gestarteten Anwendung Auflösung und Anzahl der Farben so einzustellen, daß z.B. auch Signum!2 darauf läuft.

Trotzdem sei dem Vielschreiber an dieser Stelle davon abgeraten. Kein guter Multiscan an einer der Grafikkarten kann die Bildqualität erzeugen, die die ST-Hoch- und TT-Hoch-Auflösungen mit den entsprechenden Atari-Monitoren liefern. Das liegt schlicht daran, daß ein Farbmonitor wegen seiner Lochmaske immer unschärfer sein muß. Die Bildqualität der Grafikkarten selbst ist bei (fast) allen gut, sie ist aber auch abhängig von der Kombination Monitor/Karte. Ganz entscheidend für die Beurteilung der Karten, die Sie in der Übersichtstabelle finden, waren ferner die Qualität der Einstellprogramme, der Treiber sowie der Umfang des sonstigen Zubehörs.

Im folgenden lesen Sie kurze Kritiken zu den einzelnen Grafikkarten, die Vor- und Nachteile jedes Produktes beleuchten. Betrachtet man das gesamte Testfeld, bilden sich drei Gruppen:

1. Die beiden preiswerten Karten. Die auf VGA-Systemen basierenden Crazy Dots und Imagine. Sie besitzen den bekannten ET4000-Chip von Tseng, der in der Lage ist, einfache Grafikoperationen selber durchzuführen und so erhebliche Zeitvorteile bringt. Beide sind sowohl in der 256- als auch in der 32k-Version erhältlich. Ihnen kann das beste Preis/Leistungsverhältnis zugesprochen werden.
2. Die COCO, die MEGA-Vision und die Spektrum, deren Geschwindigkeit teils niedriger ist, da sie nicht über ,intelligente' Grafikprozessoren verfügen. Sie bieten aber - mit Ausnahme der COCO - wesentlich flexiblere Grafikmodi als die VGA-Kollegen.
3. Die Spitzen'gruppe' bildet die TC1208 allein. Geschwindigkeit, Ausstattung und Qualität sind hier optimal vereint. Allerdings distanziert sich der Stolz des Hauses Matrix auch preislich klar vom Rest des Tests und weist eindeutig in den professionellen Bildverarbeitungsmarkt.

IB

Literaturhinweise:

[1] Jankowski, Rabich, Reschke: Atari Profibuch ST-STE-TT, Sybex Verlag.
[2] Augenweide Farbbild - 11 Multiscan-Monitore im Test. ST-Computer 4/92 S.20ff.

Farbe Auflösung/Farben	COCO 800x600x16	Crazy-Dots 800x600x256	Imagine 32k 800x600x256	Mega-Vision 300 800x600x256	Spektrum 1TC 800x600x256	TC-1208 800x600x256
Textausgabe [%]	92	288	311	128	125	478
Linien [%]	153	128	226	68	93	222
Rechtecke [%]	46	71	57	33	50	123
Polygone [%]	52	44	75	22	49	124
Kreise/Ellipsen [%]	116	127	184	101	35	139
Attributfunktionen [%]	253	143	488	178	227	135
Auskunftsfunktionen [%]	249	143	479	159	242	164
ESCAPE [%]	33	189	47	0	0	15
BIOS-Ausgabe [%]	53	157	47	22	*	24
GEMDOS-Ausgabe [%]	58	120	49	29	*	31
AES-Objekt-Ausgabe [%]	132	115	205	91	135	131

Monochrom Auflösung	COCO 800x600	Crazy-Dots 800x600	Imagine 32k 800x600	Mega-Vision 300 800x600	Spektrum 1TC 800x600
Textausgabe [%]	93	1150	410	125	1182
Linien [%]	62	260	168	98	265
Rechtecke [%]	73	149	143	106	159
Polygone [%]	84	229	120	24	231
Kreise/Ellipsen [%]	91	353	188	148	355
Attributfunktionen [%]	89	956	290	182	949
Auskunftsfunktionen [%]	90	684	267	161	704
ESCAPE [%]	61	190	107	0	197
BIOS-Ausgabe [%]	69	164	85	29	166
GEMDOS-Ausgabe [%]	69	597	81	38	625
AES-Objekt-Ausgabe [%]	86	313	221	122	317

100% entsprechen der Grafik-Leistung eines normalen ATARI STs

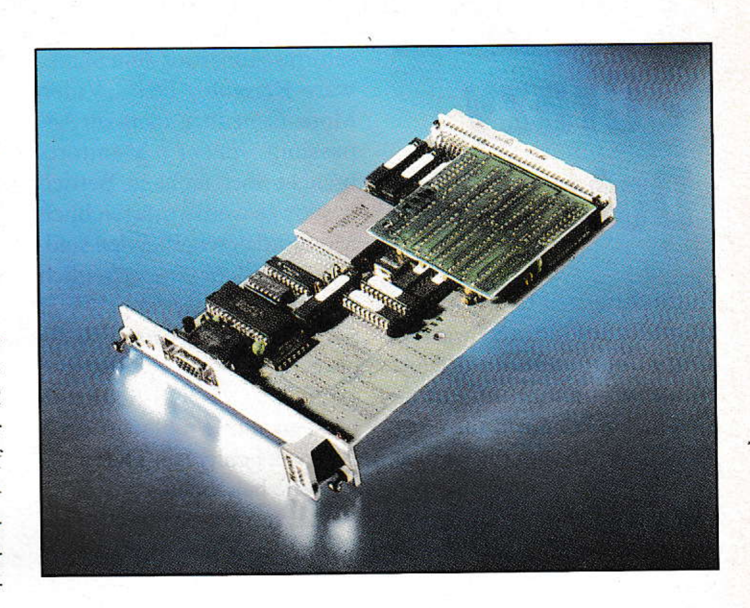
* Ausgabe erfolgte auf Originalmonitor

COCO

Das farbige Low-Cost-Angebot aus dem Hause Matrix heißt ‚COCO‘, was keinesfalls mit tropischen Früchten zu tun hat, sondern für ‚Color Controller‘ steht. Der saubere Aufbau, ein Markenzeichen des schwäbischen Herstellers, fällt sofort ins Auge. Bei der COCO handelt es sich um eine komplette Eigenentwicklung von Matrix. Der Grafikchip, der auf der Karte zum Einsatz kommt, kennt allerding nur Pixel-Frequenzen bis zu 50 MHz. Und in dieser Frequenz kann er keine 256 Farben mehr darstellen. Die nächstniedrigere Frequenz beträgt dann nur noch 28 MHz, was wiederum bei z.B. 800x600 Bildpunkten keine vernünftigen Bildwiederholfrequenzen zuläßt. Mit anderen Worten: Auflösungen mit mehr als 640x480 Punkten können bei erträglichen Wiederholfrequenzen nur mit 16 Farben benutzt werden.

Die Treiber-Software ist stabil und ausgereift, die Einstellmöglichkeiten über ein ACC sind gut. Die Erstellung neuer Monitordaten funktioniert auf sehr anschauliche Art und Weise. Es fehlt zu einem abgerundeten Bild ein Installationsprogramm, denn sogar ein Boot-Selektor ist mit in Matrix-Software integriert.

So ist die Hand-Installation ein mühseliges Unterfangen und wegen der vielen Dateien und Ordner recht unübersichtlich. Eine Deinstallation dürfte für den Laien sehr schwer sein. Das Handbuch ist dabei leider nur eine dürftige Hilfe, denn es gilt für das gesamte Produktspektrum von Matrix gleichermaßen.



Trotz der guten Qualität ist der Preis von 1099,- DM für die COCO einfach zu hoch, wenn man bedenkt, daß sie relativ langsam ist und die Fähigkeiten des TT nur unwesentlich verbessert.

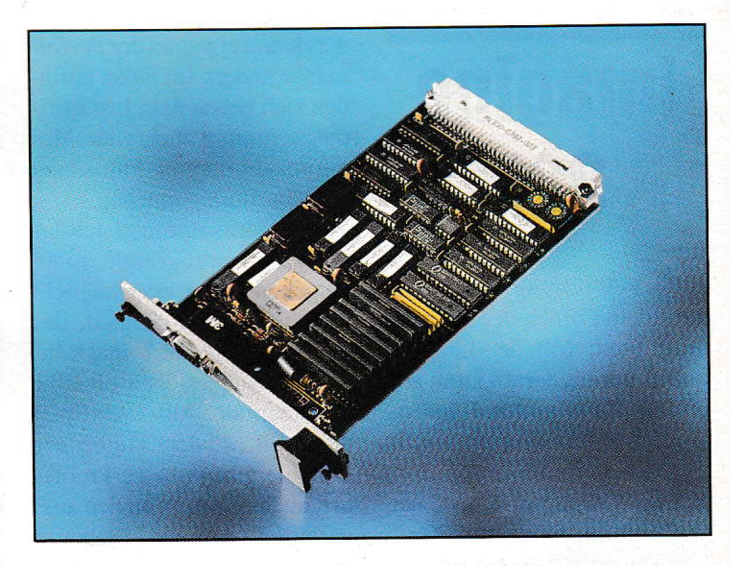
MEGA-Vision 300

Die MEGA-Vision 300 ist ein relativ neues Produkt von Sang aus Essen, das die Roßdorfer Trade iT-Leute erst seit der diesjährigen CeBIT im Vertrieb haben. Es handelt sich dabei um eine eigene Entwicklung, deren Grafikmodi bis hin zu True Color reichen.

Die MEGA-Vision 300 kommt mit nur einem einzigen Treiber im GEMSYS-Ordner aus, allerdings muß bei der Installation noch einiges andere kopiert werden. Ein Installationsprogramm gibt es leider nicht. Ebenso fehlt ein Programm, mit dem neue Monitore angepaßt werden können. Laut Entwickler ist dieses aber in Vorbereitung. Die mitgelieferten Monitordaten sind äußerst dürftig und nicht dokumentiert.

Bei den Grafikmodi bietet die MEGA-Vision ein von den anderen Anbietern abweichendes Konzept. Man definiert drei Auflösungen, wobei eine als Ersatz für die normale Atari-Auflösung gedacht ist, eine zweite der Farbauflösung entspricht, für die man die Karte verwenden will, und eine dritte eine True Color-Auflösung bietet, deren sich nur darauf vorbereitete Programme bedienen können. Dieses Konzept bleibt leider weitestgehend im Dunkeln, und die Veränderung der Auflösungen ist sehr schwierig.

Der Treiber, der keinen virtuellen Bildschirm kennt und von dem Sang momentan wöchentlich ein Update liefert, bietet leider weiteren Anlaß zur Kritik. Denn neben einer Reihe von Problemen mit ansonsten korrekt arbeitenden Programmen (GemDraw, SciGraph) ist er zum Teil quälend langsam (so z.B. das Scrollen bei Script). Mit Calamus SL hingegen gibt es keine Probleme, und dort ist auch die Geschwindigkeit erträglich. Die Entwickler verweisen auf die hohe Sicherheit, die den Treiber so bremse.



Trotz der offensichtlich leistungsfähigen Hardware hinterläßt die MEGA-Vision 300 ein ungutes Gefühl. Im ihrem momentanen Entwicklungsstadium ist sie etwas für geduldige Naturen und zudem teuer.

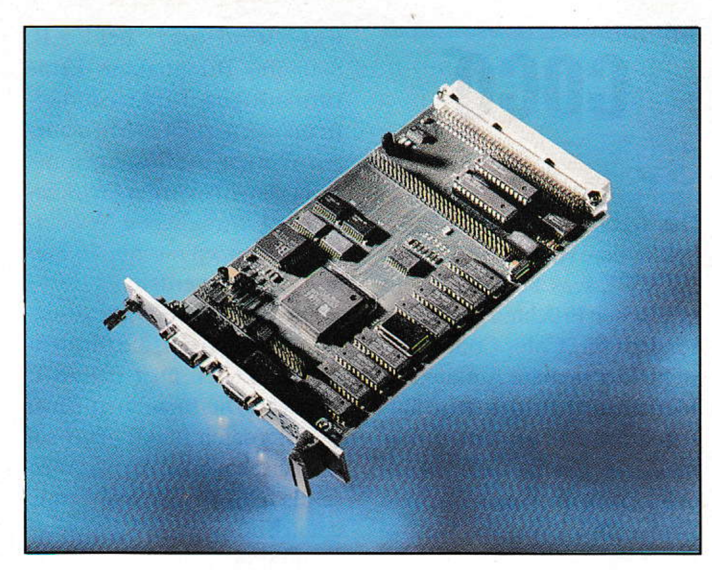
Crazy Dots 32k

Eines der ersten Grafiksubsysteme für den Mega ST war die Crazy Dots von TKR aus Kiel. Mittlerweile hat sie ein Schwesterchen bekommen, die Crazy Dots VME, um die es hier geht.

Eine besonders einfache Installation stimmt den Anwender freundlich, und das gute Handbuch steht mit Rat und Tat zur Seite: drei Treiber in den GEMSYS-Ordner kopiert, die ASSIGN.SYS angepaßt, ein Programm in den Auto-Ordner, und es kann losgehen. Mit von der Partie sind ein Boot-Selektor sowie ein Bildschirmschoner aus dem Shareware-Bereich. Der ‚Video Mode-Generator‘ dient zur Anpassung neuer Monitore, braucht aber nicht in Betrieb gesetzt zu werden, da reichlich Monitordaten vorhanden sind. Alles in allem eine erfreulich komplette Ausstattung.

Der Treiber erwies sich als ausgereifter Verteter seiner Art. Lediglich GDOS-Schriften machen ihm Probleme, doch ist dieser Fehler laut TKR mittlerweile behoben.

Die ‚verrückten Punkte‘ vertragen sich auch mit dem GDOS von NVDI, wodurch ein leichter Geschwindigkeitszuwachs zu erzielen war. Aber auch ohne NVDI gehört die Karte zu den ganz schnellen im Test, das gilt auch für den Monochrommodus. So ist sie sehr gut geeignet für alle, die mit nur einem Monitor arbeiten



wollen. Wer zunächst nur mit 256 Farben arbeiten möchte, kann seine Karte später auf 32k Farbe nachrüsten. Nach der Atari-Messe soll es eine NVDI-Version geben, die speziell auf die Farbmodi der Crazy Dots angepaßt ist. Dabei sollen laut TKR weitere Geschwindigkeitsgewinne von 200 bis 300 Prozent möglich sein.

Die Crazy Dots ist ein ausgereiftes und schnelles Produkt zu einem günstigen Preis, das vor allem Einsteigern ans Herz gelegt werden kann.

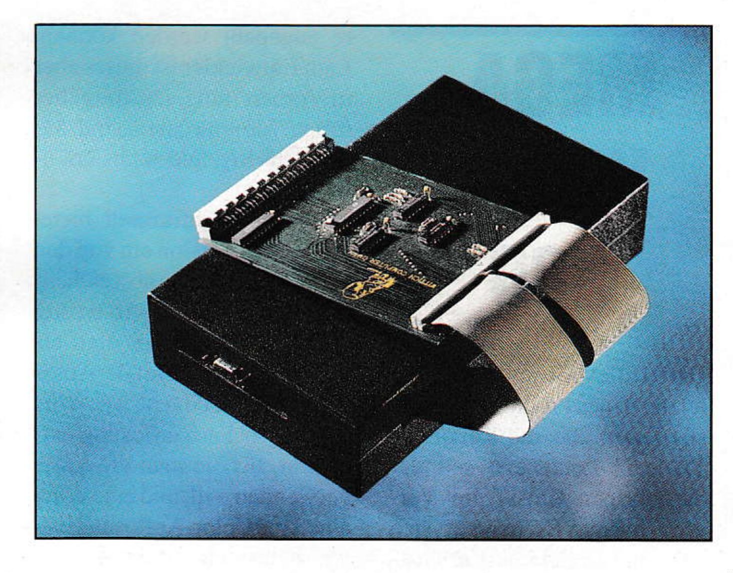
Imagine 32k

Erste Verblüffung macht sich nach dem Öffnen der Verpakkung breit: Mit der Imagine halten wir die Grafikkarte am (Flach-)Band in den Händen. Doch der Zweck heiligt halt die Mittel. Und wie der Hersteller Wittich versichert, hat er damit ein modulares Konzept geschaffen, das es ihm ermöglicht, bald noch leistungsfähigere VGA-Karten an seinen VME-Adapter anzubinden.

Die Imagine bedient sich ebenfalls des bekannten VGA-Chipsatzes von Tseng, was ihr erhebliche Geschwindigkeitsvorteile bringt. In Farbe wie in Monochrom gehört sie zu den Schnellen im Test. GDOS braucht sie nicht unbedingt, es geht auf Wunsch auch ohne, was kaum etwas an der Performance ändert. Im Paket befinden sich neben dem Installationsprogramm auch viele Monitordaten, die eine Anpassung fast überflüssig machen.

Trotzdem sollte man sich das Anpassungsprogramm nicht entgehen lassen, es ermöglicht auch dem Laien schnell, das Bestmöglichste aus seiner Röhre herauszuquetschen, ohne ihr Wohlbefinden dabei aufs Spiel zu setzen. Mit dem Treiber der Imagine gibt es keine Probleme, auch er gehört zu den ausgereiften. Auf der Testkarte waren Unschärfen zu bemängeln, die aber laut Hersteller nun nicht mehr existieren sollen.

Viel Wert haben die Wittich-Programmierer darauf gelegt, auch unsaubere Programme auf der Karte lauffähig zu machen. So liegt ihr das ‚RESWITCH.PRG‘ bei. Das Programm schaut nach, welche Applikationen gestartet wurden und schaltet entsprechend den Eintragungen in ei-



ner Steuerdatei in die richtige Auflösung und Farbanzahl. Beim Wechsel zwischen Calamus SL und Tempus Word ist somit kein Booten mehr notwendig. Für ganz harte Vertreter der unsauberen Software-Zunft gibt es einen Emulator, der permanent den Originalbildschirmspeicher auf die Karte kopiert.

Diese Features, das sehr gute Preis-/Leistungsverhältnis, der stabile Treiber und die hohe Geschwindigkeit lassen auch die Imagine attraktiv für eine Ein-Monitor-Lösung erscheinen. Prädikat: für den Einstieg bestens geeignet!

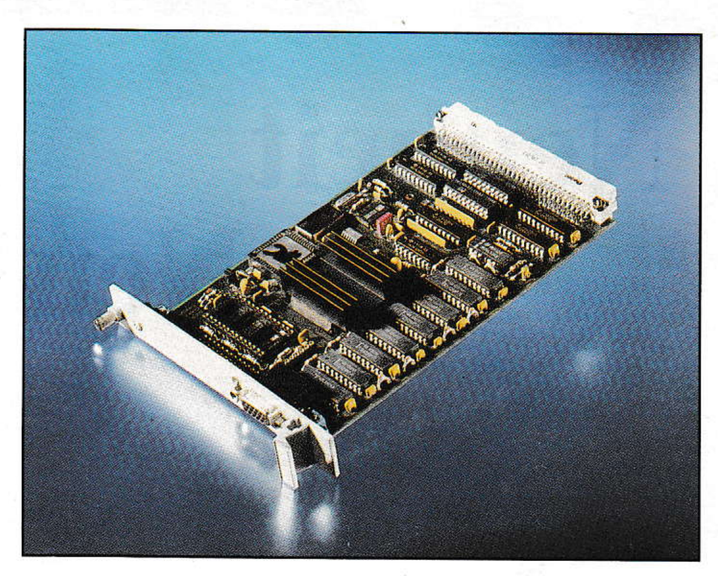
Spektrum 1 TC

Ebenfalls neu und aus westdeutschen Landen kommt die Spektrum 1 TC des Lünener Herstellers Wilhelm Mikroelektronik. Die Karte ist gleichfalls eine komplette Eigenentwicklung, und auch sie verfügt über keinen ‚intelligenten' Grafikprozessor.

Neben einem ausführlichen Handbuch gehören zum Lieferumfang der Spektrum auch ein Installationsprogramm sowie die Einstell- und Anpassungs-Software als CPX-Modul. Das sollte Schule machen. Leider ist der Platz im neuen Kontrollfeld recht beengt, so daß die Übersichtlichkeit leidet. Vermißt haben wir Monitordaten für andere als den Atari-PTC-Monitor.

Der Treiber zeigt hier und da noch Schwächen, besonders was GDOS-Schriften betrifft. Die BIOS-Textausgabe muß noch implementiert werden, sie erfolgt zur Zeit über den ROM-Bildschirmtreiber auf den normalen Atari-Monitor. Während des Tests erreichten uns neue Treiberversionen, die stets an Fehlern verloren und an Geschwindigkeit gewannen. Kurz vor Redaktionsschluß durften wir noch einen Blick auf eine Version werfen, die einige Funktionen bereits doppelt so schnell bewerkstelligt. Mittlerweile sind auch gängige Anwendungsprogramme auf den True-Color-Modus der Spektrum vorbereitet, der aufgrund des Speichers nur 640x480 Punkte groß sein kann.

Die Spektrum verfügt neben den genannten 256-, 32k- und True Color-Modi noch über



eine Gangart mit 65536 Farben, bei der volle 16 Bit genutzt werden, indem Grün sechs und Rot und Blau je fünf Bit Farbtiefe erhalten.

Von der Spektrum gibt es noch zwei kleinere Versionen: Eine verfügt lediglich über 32k Farben, die andere nur über 256. Ein Upgrade von einer zur anderen ist gegen entsprechende Zahlung möglich. Geht man davon aus, daß der Treiber in Kürze wesentlich schneller und komplettiert sein wird, ist die Spektrum 1 TC ein sehr interessantes Angebot. Vor allem wegen ihres True-Color-Modus', den man nirgendwo so preiswert bekommt.

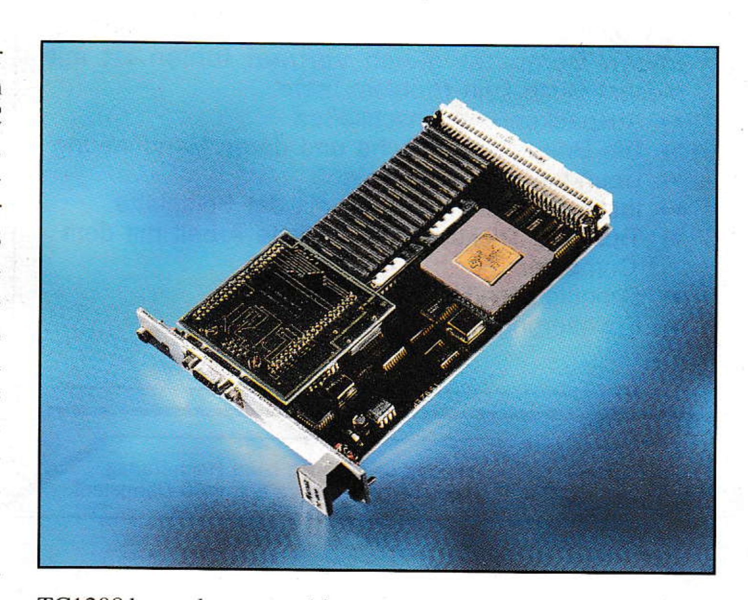
TC1208

Der Stolz des Hauses Matrix hört momentan auf den Namen TC1208. Die True-Color-Karte für höchste Ansprüche läßt die Profis unter den Bildverarbeitern nicht länger warten. Mit zwei Megabyte Videospeicher und Pixel-Frequenzen bis 132 MHz lassen sich auch Monitore von 17- und 19-Zoll-Format vernünftig ansteuern.

Die Installation der Software war bei der von uns getesteten Vorversion noch von Hand durchzuführen und dementsprechend kompliziert. Mittlerweile aber soll es laut Hersteller dafür Automatismen geben. Wie auch bei der CO-CO, so macht das Installieren neuer Video-Modi bei der TC1208 Spaß, dem Einstell-ACC ist die langjährige Erfahrung von Matrix anzusehen.

Im Betrieb zeigte der Treiber kaum Mängel, er glänzte im Zusammenspiel mit der TC 1208 durch Geschwindigkeit. Hier ist nur der 256-Farben-Modus abgebildet; es soll aber nicht unerwähnt bleiben, daß der Betrieb im True-Color-Modus nur unwesentlich langsamer ist. Mit dem Calamus SL oder Cranach ergeben sich sogar Geschwindigkeiten, die wesentlich höher liegen als bei 256 Farben, einfach weil die Software nicht mehr zu dithern braucht.

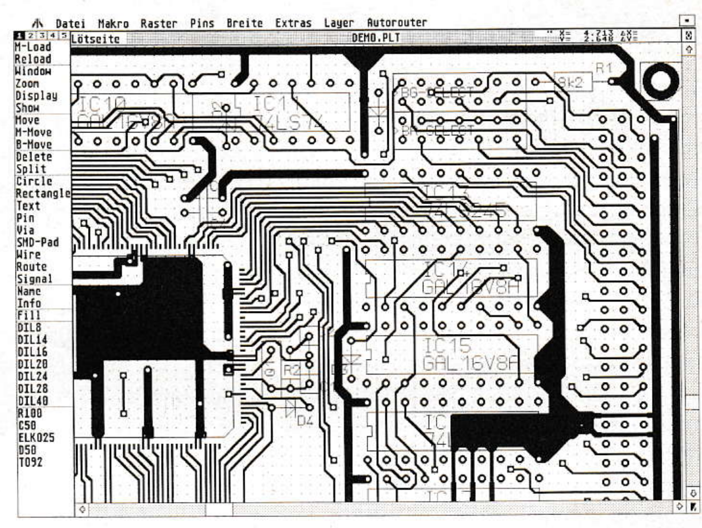
An dieser Stelle sei noch einmal erwähnt, was natürlich für alle Grafikkarten gilt: die Geschwindigkeit der BIOS- und GEMDOS-Ausgabe sowie der ESCAPES haben keinerlei Einfluß auf die Gesamt-Performance der Karten, da GEM-Programme diese Funktionen nicht nutzen. Wer ernsthaft und täglich mit großen Datenmengen aus dem Scanner zu tun hat, wird aus Geschwindigkeitsgründen nicht um Matrix'



TC1208 herumkommen. Aber nicht nur deshalb, auch wegen des ausgereiften Treibers führt für Profis kein Weg an ihr vorbei. Verschwiegen werden sollte allerdings nicht, daß für diese Karte nebst entsprechendem Monitor (19 Zoll) fast eine fünfstellige Summe veranschlagt werden muß.

Scooter-PCB

Neues Outfit für Ultimade-PCB



Das in Insider-Kreisen recht renommierte Platinen-Layout-Programm Ultimade-PCB hat einen Nachfolger bekommen. Er nennt sich Scooter-PCB und wird vom gleichen Hersteller vertrieben. Der Vorgänger zeichnete sich besonders durch sein enormes Preis-Leistungsverhältnis aus. Soviel soll schon verraten werden. Was hier mit Scooter-PCB angeboten wird, dürfte so ziemlich jeden Konkurrenten, und nicht nur im ST-Bereich, ins Schwitzen bringen...

Uns lag zum Test eine Vorabversion des neuen Scooter-PCB vor, das Ultimade-PCB noch in diesem Monat ablösen soll. Schon der Vorgänger zeichnete sich durch seine sehr große Leistungsfähigkeit aus. Vektororientierung, Netzlistenverwaltung, Autorouter, umfangreiche Drucker- und Plotter-Treiber waren alle schon in der Version 1.0 zu finden, und das zu einem extrem niedrigen Preis. Was benötigt der moderne Layouter von heute also noch?

Bevor die Frage geklärt wird, sollten wir doch einmal einen Blick zurück in die Geschichte der Platinen-Layout-Programme für ATARI-Rechner werfen. Der Urvater dürfte wohl Data Beckers Platine-ST sein, daß vor ca. 6 Jahren auf den Markt kam. Viele andere kamen und gingen auch wieder. Bei einigen handelte es sich um aufgebesserte Malprogramme, bei denen eine Leiterbahn aus vielen kleinen Pixeln bestand, andere verwendeten eine Blockrastergrafik, bei denen pro Block immer nur ein Leiterbahnsegment, ein Lötauge oder irgendein anderes Symbol möglich war. Beide Verfahren gelten heute als nicht mehr zeitgemäß, da sie den Anwender zu stark einschränken. Sie sind zwar recht einfach zu programmieren - aber was hat der Anwender davon, wenn er z.B. keinen Sub-D-Stecker auf der Platine plazieren kann, weil das Rastermaß nicht stimmt oder einen Plotter bekommt, bei dem Pixel-Grafik nicht möglich ist?

Einen völlig anderen Weg ist Scooter-PCB gegangen. Wie bei modernen Platinen-Layout-Programmen allgemein üblich, werden die Platinenlayoutinformationen vektororientiert verwaltet; das heißt, daß z.B. bei einer Linie nur die Koordinaten der Eckpunkte und die Dicke gespeichert werden, bei einem Kreis nur der Mittelpunkt und der Radius. Dieses Verfahren wird bei CAD-Programmen für Maschinenbau oder Architektur schon immer angewendet und bietet dem Anwender die Möglichkeit der einfachen Editierung der Zeichnung. Trotzdem gibt es gewaltige Unterschiede zwischen CAD-Programmen für Platinen-Layout und Maschinenbau.

Scooter-PCB benötigt beispielsweise keinen Coprozessor, um zügig arbeiten zu können, da sämtliche Daten im Integerformat vorliegen. Die kleinste verfügbare Einheit ist 1/1000 Zoll, also etwa 1/40 mm. Bei 16-Bit-Auflösung könnten damit theoretisch Platinen von einer Kantenlänge bis zu 1.6m entworfen werden.

Insgesamt stehen sieben grafische Grundelemente zur Verfügung, die jedoch vollkommen ausreichen. Dazu gehören Pins, Vias, SMD-Pads, Linien, Texte, Kreise und Rechtecke. Pins können quadratische, runde, achteckige oder längliche Formen annehmen. Der Außen- und Bohrdurchmesser kann für jeden Pin gesondert eingestellt werden. Vias sind für Durchkontaktierungen vorgesehen. Sie unterscheiden sich dadurch von den Pins, daß man ihnen keine Namen geben kann. Für SMD-Bauteile können rechteckige SMD-Pads verwendet werden, die in der Größenabmessung ebenso beliebig einstellbar sind. Texte werden durch einen fest eingebauten Vektor-Font erzeugt. Sie können natürlich gedreht und gespiegelt verwendet werden.

Ebenso wichtig wie die Grundelemente sind die Layer. Sie werden wie transparente Folien behandelt, die alle übereinander liegen und auf denen man die zuvor genannten Elemente plazieren kann. Insgesamt stehen bei Scooter-PCB 20 Layer zur Verfügung, von denen bereits 12 feste Aufgaben zugewiesen sind. So gibt es beispielsweise jeweils eine Ebene für die Platinenober- und -unterseite, für Pins und Vias, Bestückungsplan, Platinengröße oder Sperrflächen für den Autorouter. Jede Ebene ist ein- und ausblendbar. Die restlichen 8 Layer kann der Anwender frei verwenden, z.B. für Multilayer-Platinen. Für normale Anwendungen sind diese 20 Layer vollkommen ausreichend. Programme, die dem Anwender über 100 Layer anbieten, dürften ziemlich realitätsfern sein und eher der Blendung von Kaufinteressierten dienen.

Sehr gut gelöst ist die Signalverwaltung von Scooter-PCB. Ein Leiterbahnzug, der aus vielen Linienstücken, Pins und Vias besteht, wird wie eine Einheit betrachtet. Jedem Signal kann ein Name zugewiesen werden, wie z.B. GND, Vcc oder D0 bei einer Datenleitung. Wird irrtümlich ein Leiterbahnzug mit einem anderen verbunden, so daß ein Kurzschluß entstehen würde, erkennt Scooter-PCB dies und warnt den Anwender davor. Dieses Verfahren,

daß mit der Bezeichnung Online-Design-Rule-Check benannt ist, ist ein sehr hilfreiches Werkzeug, um auch wirklich eine richtig geroutete Leiterplatte zu erhalten.

Oberflächliches

Startet man das über 300 KByte große Scooter-PCB, findet man sich in der völlig neu überarbeiteten Benutzeroberfläche wieder, die einiges Revolutionäre beinhaltet. Die erste Neuerung gegenüber dem Vorgänger ist die Großbildschirmfähigkeit. Das Programm läuft jetzt problemlos mit Grafikerweiterungen wie OverScan, Pixel-Wonder oder der hohen TT-Auflösung. Einzige Voraussetzung ist der Betrieb im monochromen Modus, da Farbgrafik noch nicht unterstützt wird. Dies ist jedoch nicht allzu tragisch, da die einzelnen Layer durch unterschiedliche Füllmuster gut zu unterscheiden sind.

Altbewährt ist die Teilung der Menüs, die die Bedienung des Programmes sehr erleichtert. Häufig benötigte Funktionen befinden sich im Menü seitlich vom Arbeitsfenster und sind schnell mit der Maus erreichbar, weniger benutzte Funktionen befinden sich im Drop-Down-Menü oberhalb des Fenster. Auf die immer häufiger in Programmen aufindbaren Icons wurde hier bewußt verzichtet. Nicht etwa, um den ansteigenden Analphabetismus einzudämmen, sondern weil der Anwender die Menüs selber definieren und gemäß seinen eigenen Ansprüchen zusammenstellen kann. Wird ein Menüpunkt bei gehaltener Shift-Taste angeklickt, erscheint eine Dialogbox, wie in Bild 1 dargestellt ist. Dort befinden sich Eingabezeilen, in die der Menütitel und der Befehlstext eingegeben werden können. Wird ein Menüpunkt angewählt, dies kann per Mausklick oder über Tastencodes per Tastatur geschehen, wird der Befehlstext einem Kommando-Interpreter übergeben, der den Text auswertet und schließlich die Funktion ausführt. Wird beispielsweise der Befehl „LOAD MAC IC\DIL16.MAC" eingegeben, wird automatisch das 16polige DIL-IC als Makro geladen. Man kann sich so die häufig gebrauchten Bauteile im Menü definieren und auf Knopfdruck aktivieren. Das Seitenmenü bietet mit seinen fünf Unterspalten genügend Platz für solche Aktionen.

Erstaunlicherweise läßt sich selbst das Drop-Down-Menü auf diese Weise manipulieren, um z.B. die voreingestellten Linienbreiten den eigenen Bedürfnissen anzupassen. Selbstverständlich lassen sich alle Voreinstellungen sichern, damit sie beim nächsten Programmstart wieder verfügbar sind. Insgesamt erscheint das Verfahren sehr zukunftsweisend zu sein.

Weitere Bedienelemente sind Pop-Up-Menüs, die gelegentlich zur genaueren Spezifikation eines Befehls notwendig sind. Sie erscheinen immer unter der Maus und ermöglichen eine schnelle Auswahl.

Cache as Cache can

Um mit dem Programm arbeiten zu können, benötigt man mindestens ein Megabyte RAM, mehr kann aber nicht schaden. Nicht benötigter Speicher wird von Scooter-PCB nicht etwa brachliegen lassen, sondern als Video-Cache benutzt.

Wer schon mal mit einem vektororientierten CAD-Programmen gearbeitet hat, weiß, daß beim Neuzeichnen des Bildes Element für Element auf dem Bildschirm gezeichnet werden muß. Möchte man einen anderen Bereich der Platine sehen, muß der Bildschirm wieder komplett neu aufgebaut werden. Dieser Neuaufbau ist recht zeitintensiv und kann ganz schön stören.
Nicht so bei Scooter-PCB. Hier wird so viel wie möglich von der Platine im Speicher gezeichnet und anschließend nur ein kleiner Bereich davon in das Fenster kopiert. Beim Scrollen braucht meistens nur ein neuer Bereich aus dem Cache in das Fenster kopiert zu werden, was sehr schnell geht. Erst wenn der neue sichtbare Bereich außerhalb des Fensters liegt, erfolgt ein Redraw.

Der Redraw-Vorgang selbst ist gegenüber der Vorgängerversion, die schon recht schnell war, fast um den Faktor 10 beschleunigt worden. Wo man im PC-Bereich dem Kunden teure Grafikkarten mit intelligentem Grafikprozessor und VIDEO-RAM empfehlen würde, wurde hier einfach mit intelligenten Assembler-Routinen gearbeitet, um diesen Leistungssprung herauszuholen. Auf langsame GEM-Routinen wurde hier vollständig verzichtet. Laut Hersteller sind die verwendeten Zeichenroutinen voll und ganz auf die Bedürfnisse des Platinen-Layouts zugeschnitten. Kreise, die z.B. häufig für Lötaugen verwendet werden müssen, sind bis zu einem gewissen Durchmesser fest programmiert, so daß keine aufwendigen Berechnungen stattfinden müssen. Der Redraw des Titelbildes dieses Testberichtes hat gerade mal 3 Sekunden auf einem 8-MHz-ST gedauert. Klarer Nachteil dieses Verfahrens ist, daß noch keine Farbgrafikkarten unterstützt werden. Eine Anpassung ist aber laut Hersteller geplant.

Flächenfüllung

Eine echte Sensation ist die Flächenfüllfunktion. In dieser Qualität ist sie wohl noch nie auf einem ATARI verfügbar gewesen. Im PC-Bereich kann sie sich durchaus mit Programmen ab DM 20.000,- messen. Die Erzeugung geht für den Anwender ganz einfach vonstatten. Auf der fertig gerouteten Platine werden die Außenumrisse der Massefläche als Polygon mit der Maus definiert, ohne irgendwelche Rücksicht auf Überschneidungen mit Leiterbahnen oder anderen Objekten nehmen zu müssen. Anschließend wählt man die Flächenfüllfunktion an und gibt die gewünschten Parameter ein (Bild 2). Hier gibt man den Namen der zu füllenden Fläche an, denn Masseflächen werden wie Bauteile als Makro verwaltet. Ebenfalls wichtig sind die Angabe des Ziel-Layers, auf dem die Füllung erfolgen soll, und der Sicherheitsabstand, der mindestens eingehalten werden muß. Ferner kann ein Signal angegeben werden, mit dem die Massefläche Kontakt aufnehmen darf. Dies ist typischerweise das GND-Signal, kann aber auch im Falle einer Multilayer-Platine ein anderes Stromversorgungssignal einer Stromversorgungsebene sein. Damit Lötaugen, die voll und ganz in einer Masseflä-

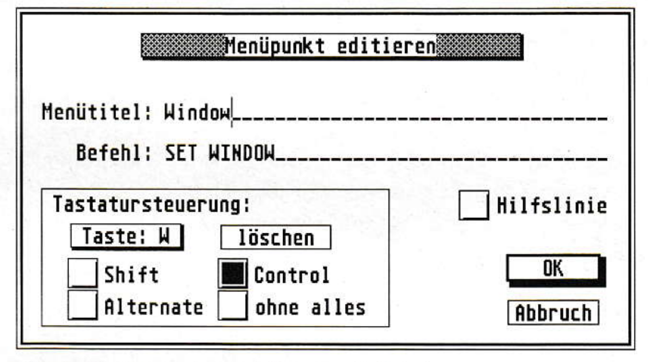


Bild 1: Die Menüs sind vom Benutzer definierbar



Bild 2: Umfangreiche Parameter zum Flächenfüllen

che stecken, besser zu löten sind, können diese wahlweise auch mit Wärmefallen versehen werden.

Im Falle von Hochfrequenzschaltungen ist es nicht nur wichtig, daß der Sicherheitsabstand genau eingehalten wird, damit der Wellenwiderstand einer Leiterbahn konstant bleibt, sondern auch, daß zwischen den Leiterbahnen keine Kupferinseln entstehen. Diese Kupferinseln haben keinen Kontakt zur Hauptmassefläche und somit auch keine abschirmende Wirkung mehr. Sie würden das Übersprechen zwischen den Leiterbahnen sogar noch vergrößern. In Niederfrequenzschaltungen stören diese Kupferinseln weit weniger. Sie sind sogar günstiger, da weniger Ätzmittel benötigt und die Umwelt mit weniger herausgelösten Kupfersalzen belastet wird. Im Programm können wahlweise Kupferinseln zugelassen werden oder nicht.

Zu guter Letzt kann das Füllmuster der Massefläche definiert werden, wobei zwischen Massiv- und Schraffurfüllung unterschieden wird. Die bei der Füllung zu verwendenden Linienstärken, -abstände und Winkel sind natürlich frei einstellbar.

Wird die Flächenfüllung gestartet, ist der Rechner je nach Prozessor und Größe der Platine eine gute Viertelstunde beschäftigt. Bei dem Umfang der durchzuführenden Berechnungen ist das nicht verwunderlich. Das Ergebnis ist jedoch erstklassig und gibt einer Platine erst das richtig professionelle Aussehen (Bild 3).

Der Autorouter

Gegenüber der Vorgängerversion ist der Autorouter nur wenig verbessert worden. Neu hinzugekommen ist das wählbare Routing-Raster, das der nach dem Lee-Algorithmus arbeitende Router verwenden soll. Mit einem 10Mil-Raster ist der Autorouter so auch in der Lage, zwei Leiterbahnen zwischen zwei IC-Pins durchzuführen. Der Sicherheitsabstand ist ebenso einstellbar wie die Breite der zu verlegenden Leiterbahnen. Auch kann jetzt einfach im Menü festgelegt werden, ob ein- oder doppelseitig geroutet werden soll. Das Anlegen von Sperrflächen entfällt damit weitgehend. Unter dem Menüpunkt „Strategie“ kann die Arbeitsweise des Autorouters festgelegt werden.

Möglich sind Angaben, ob der Autorouter einen Verlegewinkel von 90° oder gemischt 45° und 90° verwenden soll. Ein weiterer Parameter ist die Vorzugsrichtung auf den Platinenlagen. Bei doppelseitigen Platinen ist es z.B. wichtig für einen möglichst großen Routing-Erfolg, daß die einzuhaltenden Vorzugsrichtungen auf den beiden Platinenlagen verschieden sind. Bei einer strengen Einhaltung ist der Autorouter-Erfolg am größten, aber auch die Anzahl der Durchkontaktierungen.

Als Arbeitsbereiche kann der Autorouter die ganze Platine, einen rechteckigen Bereich, ein Signal oder eine einzelne Verbindung routen. Der Autorouter ist kein 100-Prozent-Router. Er kommt in der Regel zu einem Ergebnis von 80-90 Prozent, je nach Komplexität der Platine. Eine manuelle Überarbeitung der Platine ist also in der Regel notwendig. Beim interaktiven Routen erspart er dem Anwender eine Menge manuelle Verlegearbeit.

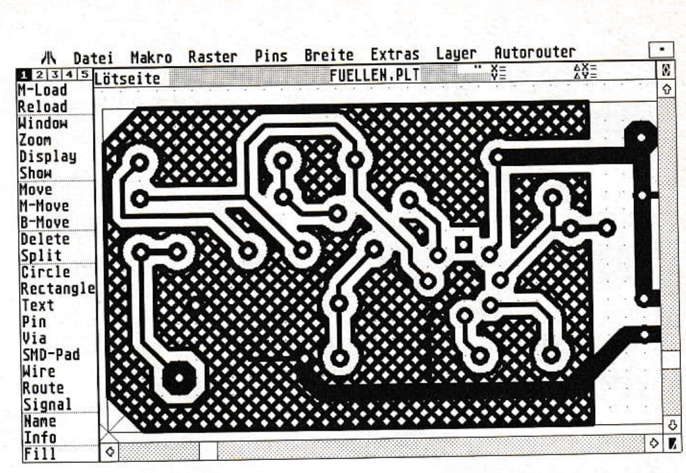


Bild 3: Flächenfüllung vom Feinsten

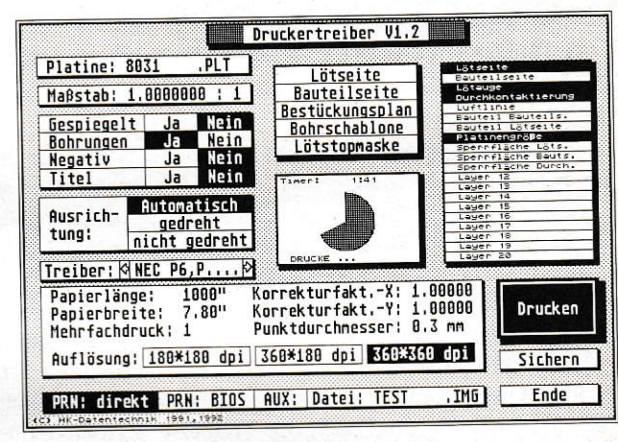


Bild 4: Der Druckertreiber läßt kaum noch Wünsche offen

Im ST-Bereich ist Scooter-PCB eines der wenigen vektororientierten Platinen-Layout-Programm mit Autorouter. Verglichen mit den restlichen Autoroutern für den ATARI, die fast alle mit dem Blockrasterverfahren arbeiten, ist das Routing-Ergebnis wesentlich besser. Besonders die freie Wahl von Routing-Raster, Sicherheitsabstand und Leiterbahnbreite ist beachtlich.

Drucken wie gedruckt

Was nützt das beste Platinen-Layout-Programm, wenn das Ergebnis nicht richtig zu Papier gebracht werden kann? Dies muß sich auch der Entwickler gedacht haben. Der hier vorhandene Druckertreiber dürf-

Durchkontaktierte Platinen

In der heutigen Zeit verliert die einseitig beschichtete Platine immer mehr an Bedeutung, da schon einfache Schaltungen mit wenigen ICs sich nicht mehr 100prozentig planarisieren lassen. Drahtbrükken auf der Platinenoberseite stellen auch nur eine begrenzte Lösung dar, da der eigentliche Sinn der Platine, alle Verbindungen zwischen den Bauteilen zu erzeugen, verlorengeht.

In der industriellen Fertigung werden doppelseitig metallisierte Platinen mit verschiedenen Verfahren hergestellt. Unterschieden wird zwischen additiven und subtraktiven Verfahren. Bei additiven Verfahren werden die Kupferbahnen chemisch auf die leere Epoxidharzplatine aufgetragen. Bei subtraktiven Verfahren wird das Leiterbahnbild durch Ätzung von vollständig kupferkaschierten Platinen erzeugt. Ein Hauptproblem stellt die Durchkontaktierung von Bohrlöchern dar. Hier hat sich die stromlose Kupferabscheidung durchgesetzt, die gegenüber galvanischen Verfahren auf kein homogenes elektrische Feld angewiesen ist, um konstante Kupferstärken zu erhalten. Der chemische und mechanische Aufwand für die Durchkontaktierung ist sehr hoch. Die Bohrlochoberfläche muß mikroporös und benetzbar gemacht werden. Ein Haftvermittler muß aufgetragen und sensibilisiert werden, damit sich schließlich ionogenes Kupfer daran abscheiden kann. Bis zur fertigen Platine sind je nach Verfahren 20 bis 30 Arbeitsschritte notwendig.

Für den Hobbyisten scheiden solche Verfahren schon aufgrund des der recht teuren, sehr giftigen und begrenzt haltbaren Chemikalien aus. Für ihn bieten sich leider nur die umständlichen Möglichkeiten an, Bauteile von beiden Seiten zu verlöten oder Kupferhohlnieten zu verwenden. Wenn man hier etwas erfinden würde, wäre sicher eine lukrative Marktlücke geschlossen. Also, fangen Sie an zu forschen!

te einige der Konkurrenten zum Schwitzen bringen (Bild 4).

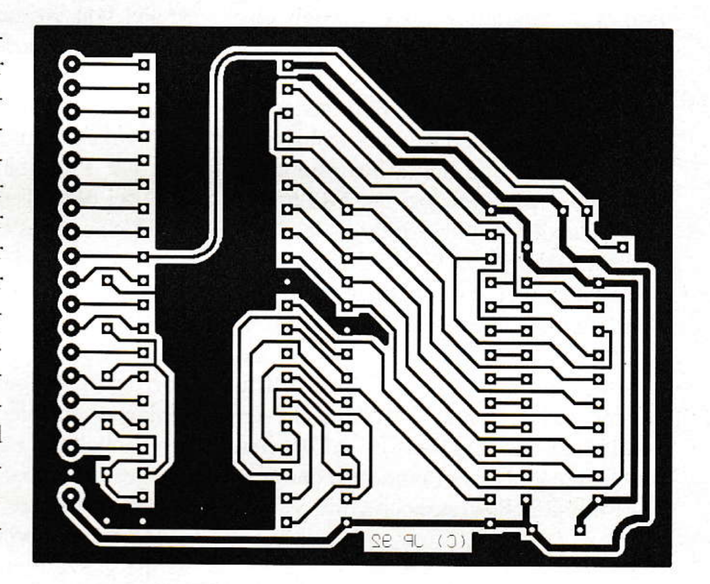
Zuerst sollte erwähnt werden, daß der Druckertreiber auf Zusatzprogramme wie GDOS nicht angewiesen ist. Dadurch entfallen aufwendige Installationen. Angeboten werden Anpassungen für nahezu jeden 9- und 24-Nadeldrucker, HP-DeskJet, HP-LaserJet und ATARI-Laserdrukker. Mit dem neuen PostScript-Treiber können jetzt auch PostScript-Geräte wie Laserdrucker oder Filmbelichter unmittelbar angesteuert werden. Ebenfalls neu ist ein Treiber, mit dem GEM-Image-Dateien von der Platine erzeugt werden können. Diese IMG-Dateien können heute von fast jedem Textverarbeitungsprogramm eingelesen werden. Auch die Ausgabe vom vektororientierten Metafile-Format ist möglich, so daß eine Übergabe an Programme wie z.B. Calamus möglich ist.

Der Ausgabemaßstab ist stufenlos einstellbar. Mechanische Toleranzen beim Drucken können durch Korrekturfaktoren behoben werden. Selbst der Nadeldurchmesser wird berücksichtigt, um Leiterbahnen in der gewünschten Breite zu bekommen. Bei schwachen Farbbändern kann eine Druckzeile auch mehrfach bedruckt werden. Für jeden Drucker stehen meist verschiedene Auflösungen zur Verfügung, je nachdem, ob ein schneller Probeausdruck oder ein hochwertiger Ausdruck stattfinden soll. Selbstverständlich können nicht nur die Platinenlagen gedruckt werden, sondern auch Bestückungspläne, Bohrschablonen und Lötstopmasken generiert werden.

Der Plotter-Treiber...

...unterstützt folgende Datenformate: HPGL, CAlComp, Gerber-Foto-Plotter und Excellon-Bohrautomaten. Bei den Stift-Plottern wird Wegoptimierung, freie Wahl der Stiftdurchmesser und stufenloser Ausgabemaßstab unterstützt. Neu hinzugekommen ist eine Stiftoptimierung, bei der zur Beschleunigung der Ausgabe immer der größtmögliche Stift zur Zeichnung eines Elementes verwendet wird. Zur Kontrolle des Layouts ist auch ein Farb-Plot möglich. Hier wird für jede Ebene eine andere Farbe verwendet. In der industriellen Leiterplattenfertigung werden meistens Foto-Plotter verwendet, die mit Gerberdateien gesteuert werden. Auch diese werden unterstützt. Beim Foto-Plotter fällt ein Lichtstrahl durch eine veränderliche Blende und erzeugt auf einem Film das Leiterbahnbild. Die Vergabe der Blendennummern erfolgt manuell in der Aperturtabelle, so daß auch Foto-Plotter mit fest vergebenen Blendennummern unterstützt werden.

Ebenfalls werden in der professionellen Leiterplattenfertigung CNC-Maschinen zur Bohrung der Platinen verwendet. Auch das häufig verwendete Excellon-Format, welches in DIN-Norm-66025 definiert ist, wird von Scooter-PCB unterstützt.



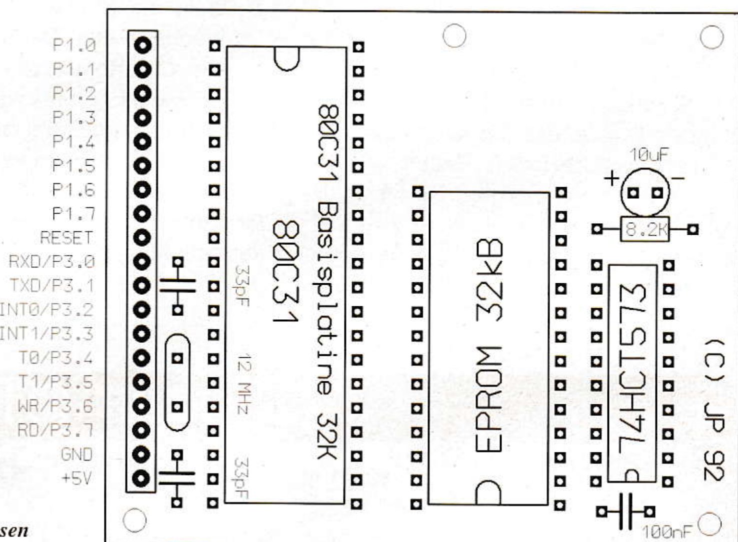


Bild 5: Der Output kann sich sehen lassen

Praxistest

Um die Praxistauglichkeit des Programms zu testen, haben wir uns entschlossen, ein kleines 8031-Mikro-Controller-Board zu entflechten, das für universelle Aufgaben eingesetzt werden kann.

Damit das Platinen-Layout-Programm weiß, welche Verbindungen hergestellt werden sollen, muß eine Verbindungsliste erzeugt werden. Hierfür gibt es mehrere Möglichkeiten. Die bequemste ist sicherlich die Erzeugung mittels Schaltplanprogramm. Wer z.B. Zugang zu dem PC-Schaltplanprogramm OrCAD-SDT hat, kann es problemlos verwenden. Laut Hersteller soll noch Ende diesen Jahres ein Schaltplanprogramm für den ATARI folgen.

Alternativ kann auch mit einem Texteditor sehr einfach eine Verbindungsliste erstellt werden. Mit einem kleinen Übersetzungsprogramm wird dieser Text dann in ein Netzlistenformat überführt, das von Scooter-PCB lesbar ist. Für die Beschreibung dieser Platine waren so lediglich 11 Zeilen Text notwendig.

Wenn der Text fehlerfrei übersetzt wurde, kann mit dem Platinen-Layout begonnen werden. Zuerst werden die benötigten Bauteile aus der mitgelieferten Bibliothek geladen und provisorisch auf der Platine abgelegt. Dann gibt man den Bauteilen ihre Namen und Bauteilwerte. Jetzt kann die Verbindungsliste eingelesen werden. Die Bauteile werden mit dünnen Luftlinien verdrahtet, die sich beim Verschieben der Bauteile wie Gummibänder mitbewegen. Wenn eine vorerst optimale Bauteillage gefunden ist, kann mit dem Routen begonnen werden. In diesem Fall haben wir uns vorwiegend für das manuelle Routen entschieden, um ein einseitiges Platinen-Layout ohne jede Brücke zu erzeugen. Kurze Stücke wurden vom Autorouter verlegt.

Das manuelle Routen geht sehr einfach. Man klickt einfach ein Ende einer Luftlinie an und verlegt Leiterbahnen mit der Maus, bis das andere Ende der immer mitgeführten Luftline erreicht ist. Mit der rechten Maustaste kann dabei der Knickwinkel für Leiterbahnen festgelegt werden, wobei auch kreisförmige Leiterbahnzüge möglich sind.

Leiterbahnen oder Bauteile, die nicht richtig liegen, können nachträglich problemlos verschoben werden, um Platz für andere Leiterbahnen zu schaffen. Die angeschlossenen Leiterbahnen werden dabei ebenfalls wie Gummibänder mitgezogen. Insgesamt kann das Handling als sehr gelungen bezeichnet werden.

Zur Herstellung der Platine gibt es mehrere Möglichkeiten. Hat man einen Nadeldrucker, druckt man die Platine am besten im Maßstab 2:1 aus und läßt sich die Zeichnung im nächstgelegenen Copyshop auf Folie oder besser Transparentpapier kopieren. Besitzer von Laserdruckern haben es einfacher. Sie können direkt auf solche Materialien drucken und damit ihre Platinen belichten.

Auch die industrielle Herstellung von Platinen ist mit Scooter-PCB möglich: Die Treiber von Scooter-PCB unterstützen alle notwendigen Datenformate einer vollautomatischen CNC-Leiterplattenfertigung. Da jeder „PC" ATARI-Disketten lesen kann (was leider nicht jeder weiß), ist der Datentransfer ebenfalls problemlos. Wir hatten in letzter Zeit wiederholt Gelegenheit, mit Scooter-PCB entworfene Leiterplatten fertigen zu lassen, das Ergebnis war durchweg professionell.

Fazit

In diesem Testbericht konnte wirklich nur das Wichtigste erwähnt werden. Zu vielfältig sind die Möglichkeiten, die Scooter-PCB bietet. Ein Converter, der alte „Platine-ST"-Dateien übersetzt, blieb ebenso unerwähnt wie die Möglichkeit, eigene Bauteile zu erzeugen. Unschlagbar an diesem Programm ist ganz bestimmt das Preis-Leistungsverhältnis. Mit allen zuvor beschriebenen Eigenschaften wie Autorouter, Druckertreibern, Plotter-Treibern, Bauteilbibliotheken und Übersetzungsprogrammen kostet das Komplettpaket 279,- DM. Mit der konsequent verwendeten Vektororientierung ist einfache Bedienbarkeit auch bei anspruchsvollen Platinen gewährleistet. Positiv ist auch die Stabilität, mit der schon die Vorabversion des Programmes lief. Ernsthafte Fehler oder gar Abstürze waren nicht feststellbar. Sicherlich gibt es auch Schwachstellen in diesem Programm, wie z.B. eine fehlende Undo-Funktion. In manchen Funktion ist Scooter auch nicht gerade schnell. Wer innerhalb kurzer Zeit ein 486er-Board entflechten muß, weil er sein Geld damit verdient, sollte also vielleicht doch lieber bei 20.000,- DM-Programmen bleiben. Für alle anderen Anwender kann man Scooter-PCB uneingeschränkt empfehlen!

Jürgen Piscol

Bezugsquelle:

HK-Datentechnik
Hubert Kahlert
Heerstraße 44
W-4047 Dormagen 11
Tel.: (02133) 91244

GFA-BASIC nach C Konverter Version 3.1 professional



Verwandlungs-künstler

Vielen GFA-BASIC-Programmierern droht der Anschluß an aktuelle Entwicklungen wie MultiTOS, MiNT, FSM-GDOS, Grafikkarten und neue ATARI-Modelle durch prinzipielle Einschränkungen von BASIC sowie mangelnde Pflege von GFA-BASIC für den ATARI verlorenzugehen. Auch die Umsetzung auf andere Rechnersysteme gestaltet sich schwierig.

Neben den Fehlern in GFA-BASIC selbst wird auch durch die gegenüber anderen prozeduralen Sprachen wie C, Pascal oder Modula fehlenden Sprachelemente wie abstrakte Datentypen und Modulkonzepte mit zunehmender Programmgröße die Entwicklung schwieriger. Um hier einen Ausweg zu bieten und eine zeitraubende Neuentwicklung in C zu ersparen, dient der „GFA-BASIC nach C Konverter" von Cicero.

Lieferumfang

Der Konverter selbst läuft im GEM-Gewand in beliebigen Auflösungen ab ST mittel auf allen ST/STE/TT-Modellen und benötigt einen RAM-Ausbau von 1 MB, bei größeren Projekten besser mehr.

Neben Programm- und Bibliotheksdiskette wird ein fast 500 Seiten starkes Handbuch im DIN-A5-Ringordner mitgeliefert, das über einen ausführlichen Referenzteil und Index verfügt. Zwar ist die Schrift zum Teil recht klein geraten, es fehlen Grafiken, und an manchen Stellen wäre eine klarere Strukturierung im sehr technisch formulierten Handbuch wünschenswert, doch werden Problembereiche bei der Konvertierung sowie Funktionen des Konverters meist klar und umfassend dargestellt.

Anwendung

Es gibt zwei Zielsetzungen des Konverters, zum einen die Weiterentwicklung von Programmen in BASIC, wobei die Konvertierung aus Gründen der besseren Codeerzeugung und Fehlerprüfung von C-Compilern, ggf. Ausnutzung einer FPU sowie Portierbarkeit, erfolgt, zum anderen die endgültige Umsetzung, bei der der C-Quelltext die Grundlage für die weitere Entwicklung sein soll.

Um das erste Ziel zu erreichen, existieren derzeit Bibliotheken im Turbo- bzw. Pure-C-Format (für Lattice C in Vorbereitung), die den Befehlssatz von GFA-BASIC in C auf dem ATARI nachbilden sowie entsprechende für MS-DOS und Windows. Weitere sind für den NeXT und Apple MacIntosh in Arbeit. So lassen sich auf der Grundlage eines GFA-BASIC-Ausgangsquelltextes Programme für verschiedene Rechnerfamilien pflegen, wobei neben der Programmstruktur auch einheitliche Grafikbefehle und Menüs zur Verfügung stehen und man weiterhin Gebrauch von der einfachen und mächtigen Sprache GFA-BASIC machen kann.Mehr dazu im zweiten Teil des Testberichtes.

Unterstützt wird dieses Ziel noch durch die Konstruktion abstrakter Datentypen, die in GFA-BASIC als einzelne Felder definiert, markiert und vom Konverter im C-Source zu echten Datenstrukturen zusammengefaßt werden (s. Beispiel).

Die zweite Einsatzmöglichkeit zur Erzeugung eines weiterzupflegenden C-Quelltextes erfordert einige Nacharbeit, denn auf das spätere Aussehen kann bis auf den Klammerungsstil bei Anweisungsblöcken kaum Einfluß genommen werden. Bei der Beeinflußbarkeit solcher stilistischen Dinge, die durch die bessere Lesbarkeit eine einfachere Wartung ermöglichen würden, will Cicero aber einen Schwerpunkt zukünftiger Versionen setzen.Unterstützung bei der Vielzahl an Bibliotheksfunktionen erhält man durch die in Pure C einzubindende Online-Hilfe.

Modularisierung

Der Konverter bietet eine automatische Modularisierung an, die den BASIC-Source in einzelne Module mit den entsprechenden Schnittstellen für Variablen und Funktionen unterteilt. Noch erfolgt die Trennung anhand der einstellbaren Zeilenanzahl, es soll jedoch einmal die Möglichkeit geben, im GFA-Source Trennstellen anzugeben.

Wie schon bei kompletten Programmen werden sämtliche Module und Bibliotheken in eine Turbo-/Pure-C-Projektdatei eingetragen, so daß einem sofortigen Compiler-Lauf nichts im Wege steht.

Aufwand

Nun die entscheidende Fragestellung, nämlich nach dem notwendigen Aufwand, um aus dem BASIC Quelltext ein lauffähiges Äquivalent in C zu erhalten. Der Konverter unterstützt, von wenigen Ausnahmen wie z.B. EVERY/AFTER und RESUME abgesehen, sämtliche Befehle von GFA-BASIC 3.0.

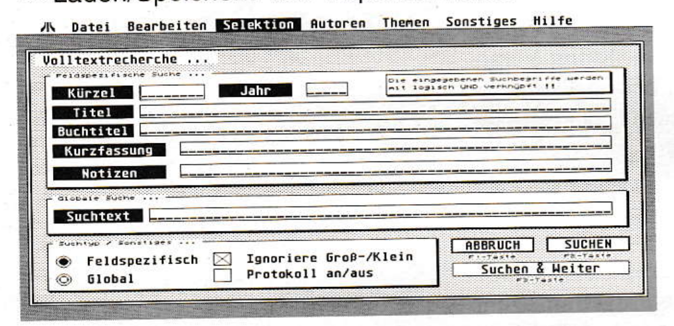
Zunächst einmal gilt es, einige sprachbedingte Unterschiede zu erkennen und den Source ggf. anzupassen, wobei das Handbuch sehr gute Hilfestellung bietet. Bei der Vielzahl an Befehlen und zum Teil im GFA-BASIC Handbuch nicht dokumentierten Optionen und Parametern kam es vor, daß der Konverter einige nicht korrekt übersetzen konnte. In diesen Fällen konnte Cicero innerhalb kurzer Zeit

SOFTWARE-HIGHLIGHTS

Review 2.1

Das Literatur-Archiv und Recherche-System für alle Atari ST und TT Computer.

- Menü- und Tastatursteuerung, Pull-Down Menü
- Relationales Datenbanksystem mit Dateneingabe in fertige Masken, Vollständigkeitsüberprüfung und automatische Vorbelegung der Felder: Numerierung, Datierung, Sortierung
- Datenimport und Export (Laden und Speichern)
- Volltextrecherche (global/lokal), Katalogsuche
- Laden/Speichern von Reportformaten



- Reporte in beliebig gestaltbaren Formaten
- Installierung auf einem Zentralrechner ist möglich, Datenbestand auf dem Hostrechner Datentransfer via Netzwerk
- Läuft auf allen Atari ST und TT und einem monochrom-Monitor

Neu ab der Version 2.1 im Programm: erweiterter Dateneditor, erweiterter Export- und Report-Editor, erweiterte Volltextrecherche, Vereinfachung des Exportformats, Protokoll-Verwaltung, Export-Konverter, Import von Journalkatalogen, Verbesserte Unterstützung von Netzwerken, Direkte Selektion aller neu erfaßter Zitate, Optimierte Dateneingabemasken

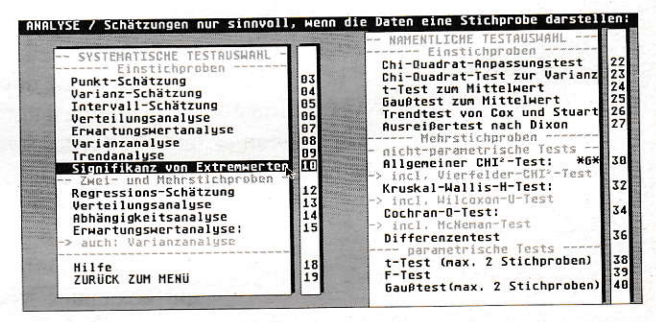
248,- DM

Preise sind unverbindlich empfohlene Verkaufspreise

Statistik Profi

Ein Programm für alle Wissenschaftler, Marktforscher, Statistiker und Studenten.

- Zahlreiche analytische & deskriptive Verfahren
- Vollautomatische Grafikdarstellung aller geeigneten Verfahren
- Import/Export-Funktionen für Datentransfer



- Multivariate Verfahren mit umfangreichen Grafikfunktionen
- Anwenderfreundliche Bedienung und umfangreiche Hilfsfunktionen
- Umfangreiche integrierte Servicefunktionen, wie automatische Fragebogenerstellung
- Kapazitätsbegrenzung nur durch externen Massenspeicher
- Eigene Programmiersprache CCL für komplexe Manipulationen
- Modul-Erweiterungsmöglichkeiten
- Integrierter Grafik- und Texteditor
- Namhafte Institute (z.B.: AfiB in Heidelberg) zählen zu den Anwendern.

248,- DM

Hiermit bestelle ich beim Heim-Verlag, Heidelberger-Landstr.194, 6100 Darmstadt

____

____

____

Ich zahle (zuzüglich 6,- DM Versandkosten Ausland 10,- DM) unabhängig von der bestellten Stückzahl

- ❑ per Nachnahme
- ❑ per Scheck

Name : ______________________

Straße: ______________________

Ort : ______________________

Heim Verlag GmbH

Heidelberger-Land-Str. 194
6100 Darmstadt-Eberstadt
Tel.: 0 61 51 / 5 60 57 Fax: 5 60 59

eine angepaßte Version bereitstellen. Generell erleichtert eine strukturierte, saubere Programmierung in BASIC die Übersetzung erheblich, denn sowohl Konverter als auch ANSI-C-Compiler sind viel genauer in Syntaxprüfungen als GFA-BASIC. Trotzdem kommt es durchaus noch zu fehlerhaft übersetzten Stellen, die dann dem C-Compiler auffallen. Hier ist dann eine meist recht einfache Nachbearbeitung, die natürlich Kenntnisse in C voraussetzt, notwendig.

Wer erwartet, nach der Konvertierung ein auf Anhieb lauffähiges Programm zu bekommen, mag vielleicht enttäuscht sein, bei all dem darf der Hauptvorteil des Konverters jedoch nicht vergessen werden, nämlich dem Programmierer in kurzer Zeit zu einem C-Source zu verhelfen, ohne daß eine monatelange Neuentwicklung oder Übersetzung von Hand nötig wird. Neben den zu Anfang angeführten Vorteilen wird damit vor allem die weitere Pflege des Produktes nicht verzögert. Auf Anfrage werden dabei von Cicero auch Programmportierungen sowie Schulungen durchgeführt.

Strings

Ein von vielen BASIC-Programmierern geschätzter Vorteil von BASIC ist dessen dynamische String-Verwaltung, die in GFA-BASIC jedoch mit dem eklatanten Nachteil einer fehlerhaften und unflexiblen Speicherverwaltung verbunden ist, der die Nutzung von MiNT und MultiTOS sowie auf TTs erheblich erschwert.

Hier liegt ein Schwerpunkt der Bibliotheken, so werden sowohl eine statische String-Verwaltung, bei der jedem String ein Speicherbereich bestimmter Größe zugeordnet wird, als auch eine dynamische, die bei String-Vergrößerungen selbsttätig Speicher anfordert, angeboten, wobei sich auch Abweichungen bei bestimmten Strings angeben lassen. Bei Speichermangel erscheint eine Warnmeldung, so daß ein Programmabsturz in den meisten Fällen vermieden wird. Bei der maximalen String-Länge kann man noch zwischen 64 KB und 2 GB wählen, wichtig für Befehle wie BGET und BPUT bei höheren Grafikauflösungen.

Die Konvertierung

Der Konverter übersetzt auf einem 16-MHz-ST gut 2000 Zeilen/min, wobei Fehlermeldungen und Warnungen optional in eine Datei geschrieben werden. Ärgerlich ist aber, daß jede Fehlermeldung von Hand zu quittieren ist, selbst wenn der Konvertierungsvorgang fortgesetzt werden kann. Erfreulicherweise sind die Bibliotheken frei von LineA-Aufrufen, selbst bei den Befehlen, in denen GFA-BASIC darauf zurückgreift. Diese Umstellung von LineA auf VDI kommt im Konvertat vor allem der Lauffähigkeit auf Grafikkarten zugute.

Lediglich die Initialisierungsfunktion ist aus Gründen der Kompatibilität zu GFA-BASIC etwas unsauber, was allerdings bei Programmen, die nicht davon abhängig sind, kein Problem darstellt, da der Quelltext erfreulicherweise mitgeliefert wird und Änderungen darin einfach sind.

Ergebnisse

Konvertierte Programme sind generell etwas größer als die entsprechenden GFA-BASIC-Pendants, vor allem bei kurzen Programmen. Die Geschwindigkeit ist je nach Einstellung des Konverters bei ganzzahligen Berechnungen und String-Verarbeitung meist doppelt, bei Fließkommaberechnungen unter Nutzung der FPU um ein Vielfaches schneller.

Testobjekte

Als größere Testobjekte kamen zum einen ein bekanntes Mailbox-Programm, zum anderen CatSieve, ein sauberes GEM-Programm von Uwe Ohse, mit einem Umfang von 450 bzw. 250 KB, zum Einsatz. Das Mail-Poxprogramm konnte innerhalb der Testzeit immerhin ansatzweise lauffähig gemacht werden, bis sich herausstellte, daß noch einige Anpassungen am GFA-Source nötig sind, während es beim GEM-Programm derzeit noch einige Konverter-Fehler gibt, allerdings handelt es sich dabei auch um einen mir wenig vertrauten Quelltext.

Kleinere Programme mit etwa 2500 Zeilen, Grafik und Assembler-Routinen waren innerhalb von etwa zwei Tagen weitgehend lauffähig. Den Aufwand für das Mailbox-Programm kann man auf etwa drei Wochen veranschlagen, während eine Neuentwicklung oder Konvertierung von Hand minimal sechs Monate bis ein Jahr in Anspruch nähme.

Resümee

Mit dem „GFA-BASIC nach C Konverter“ erhält man ein Paket, das sowohl aus der Software als auch dem Service dahinter besteht, wobei man das Gefühl hat, ein aktives, ständig weiterentwickeltes System vor sich zu haben und bei Fragen und Problemen schnell Hilfestellung erhält.

Frank Baumgart

Cicero Innovative System-Software
Ballweilerstraße 7
6676 Mandelbachtal
Tel.:(06803) 2834

Preise:

BASIC nach C Pionier	*189,-*
BASIC nach C Professional	*449,-*
BASIC nach C MS-Windows	*549,-*

Literaturhinweise:

Christoph Conrad
- LineA-freie GFA-BASIC-Library

Uwe Ohse
- Liste des Grauens: Bugs in GFA-BASIC
- Bugsic MiNT-Fix

Frank Baumgart
- Setscreen-Fix: GFA-BASIC auf Großbildschirmen

zumindest die beiden ersten Texte sind in vielen Mailboxen des Mausnets zu finden (z.B. QUARK-Paderborn 05251/71409).

```
$SA
DIM struct.person.vorname$(10)
DIM struct.person.nachname$(10)
DIM struct.person.alter&(10)
DIM struct.person.groesse#(10)
$SE
```

Listing 1: Datenstrukturen in GFA-BASIC...

```
struct
  {
  Feld    vornameS_f;
  Feld    nachnameS_f;
  int     alterI[10+1];
  double groesseD[10+1];
  } person;
```

Listing 2: ...und deren Konvertierung in C

```
 B NACH C  Datei  Konvertierung  Module  Optionen  System
                          P:\RISIKO.LST
MOD:LST-C KON:Ansi TYP:Por.h OPT:Alle  UMW:zeig  END:kurz WAR:lang UEB:Prg KF:3
    ' Ausgabe des Gewinners
    ALERT 1,spieler$(spieler)+"|hat gewonnen !",1," OK ",button
    '
    PUT 0,0,map$
    '
    ALERT 2,"Möchten Sie ein|neues Spiel",1," ja |nein",button
  UNTIL button=2
  IF u=2
                          P:\RISIKO.C
     /*  Ausgabe des Gewinners */
   B_alert( 1,B_strcat(spielerS_f.p.S2[spielerW],"|hat gewonnen !"),1," OK "
     /*  */
   B_put(0,0,mapS,-1);
     /*  */
   B_alert( 2,"Möchten Sie ein|neues Spiel",1," ja |nein",buttonW);
 }  while(!((buttonW==2)));
 if((uW==2))
```

Gegenüberstellung von BASIC- und C-Source im Konverter



Der ATARI ist ein Allround-Künstler, wenn es um Betriebssysteme geht. Kaum ein anderer Rechner kann mit so vielen Betriebssystemen begeistern wie er. Dennoch ist dieser Rechner in vielen Bereichen (z.B. im kaufmännisch-wirtschaftlichen Bereich) leider nicht so vertreten, wie man es sich wünschen würde. Aus diesem Grunde war ich gezwungen, Programme, die ich auf dem ST entwickelt hatte, auf einen PC zu übertragen. Mit Freude hörte ich daher von GFA Systemtechnik, daß meine Sprache nun auf MS-DOS und sogar auf Windows übertragen worden ist. In Zeitschriften las ich viel Positives - also die Lösung meiner Probleme?

MS-DOS-Tools, WINI-Tools

Konvertierung nach Windows

Bestellt - bekommen - probiert - frustriert, so kann man schnell meine Erfahrungen mit dem GFA-BASIC für den PC beschreiben. Obwohl mittlerweile bessere Releases der Implementierungen für Windows und MS-DOS auf dem Markt sind, muß ich sagen, daß es nahezu unmöglich ist, ATARI-GFA-Programme unverändert auf den PC zu übertragen. Kaum ein Grafikbefehl, der die gleichen Eigenschaften wie auf dem ATARI hat. Kein schlechtes BASIC - doch keine Lösung, geschweige denn eine betriebssystemunabhängige Programmiersprache.

Der BASIC-Konverter nach C für Microsoft Windows

Nachdem ich auf der ATARI-Messe 1991 hörte, daß es CICERO Innovative System-Software gelungen ist, über den direkten Weg der BASIC-Konvertierung nach C Programme auf den PC zu übertragen, wagte ich noch einen Versuch: Also bestellte ich den BASIC-Konverter nach C inklusive MS-DOS- und WINI-Tools, die für PCs ohne Windows oder mit Windows die Lauffähigkeit meiner Programme erzeugen sollten.

Bedienung und Oberfläche ...

... des Konverters sind das genaue Abbild des ATARI-Konverters, jedoch die Möglichkeit, auf die Konvertierung Einfluß zu nehmen ist wesentlich größer. Dieser Konverter versteht eine sogenannte Initialisierungs-Datei. In dieser lassen sich z.B. eigene Funktionen und Konstanten angeben, Befehle aus der Source herausnehmen, Parameterstellen von Funktionen und Such-Strings festlegen. Sogar die Umdefinition der Variablentypen durch Neuzuweisung der Endungen oder zusätzliche Zeilenende-kennungen wie z.B. der von anderen Dialekten gewohnte ‚:‘ ist möglich - eine sehr flexible und offene Konvertierungshilfe. Zum Spaß habe ich mir in 10 Minuten einen Pascal-nach-C-Konverter zurechtprogrammiert. Eine Eigenschaft dieses Konverters ist noch besonders zu erwähnen: Schreibt man hinter REM den Ausdruck: „BNC-GFA:“, kann man Code hineinschreiben, der zwar von GFA-BASIC überlesen, vom Konverter jedoch übersetzt und übernommen wird. Schreibt man stattdessen hinter REM den Ausdruck „BNC-C:“, fügt der Konverter den dahinter stehenden C-Code direkt ohne Übersetzung in die Source ein. Durch diesen Trick lassen sich viele Teile, die nur in C möglich sind, in die GFA-Source einfügen, ohne daß sich der Interpreter beschwert.

```
fuellen
mouse
texte
PROCEDURE fuellen
  CLS
  DEFFILL 3,2,6
  PBOX 0,0,_X,_Y
  FOR j&=0 TO 10
    DEFFILL j&,3,3
    PBOX 10,10,30,30
    GET 10,10+19,30,30+19,a$
    FOR i&=0 TO 15
      PUT 30+i&*50,10+19+30*j&,a$,i&
    NEXT i&
  NEXT j&
  KEYGET a%
  CLS
RETURN
PROCEDURE mouse
  CLS
  PRINT "Mouse-Zeiger - Bitte Taste drücken"
  FOR i&=0 TO 7
    PRINT AT(1,2);"Mouse-Zeiger Nr.:"'i&;"  "
    DEFMOUSE i&
    KEYGET a%
  NEXT i&
  DEFMOUSE 0
  CLS
RETURN
PROCEDURE texte
  LOCAL test$,farbe&,art&,winkel&,groesse&
  text$="Hello BASIC nach C"
  FOR farbe&=1 TO 15
    FOR groesse&=0 TO 20
      DEFTEXT farbe&,art&,winkel&,groesse&
      TEXT 0,20+20*groesse&,text$
    NEXT groesse&
    KEYGET a%
    CLS
  NEXT farbe&
RETURN
```

Ein GEM-Sourcecode in GFA-BASIC vor ...

Grafik?

Stichwort: Betriebssystemunabhängige (Grafik-)Programmierung CICERO Innovative System-Software proklamiert ein hehres Ziel, auf dem ATARI in BASIC oder C zu programmieren und die Programme auf andere Betriebssysteme wie MS-DOS, Windows, UNIX und Apple übertragen zu können. Doch die Behauptung kannte ich schon von anderen Firmen. Lassen sich also auf dem ATARI in C erstellte Programme oder konvertierte GFA-BASIC-Programme auf MS-DOS-Rechner oder nach Windows übertragen?

Zum Test stand mir die Version 3.1 des Windows-Konverters und 1.1 der ATARI-, MS-DOS- und WINI-Tools zur Verfügung. Generell: C-Programme, die mit den ATARI-Tools, dem Pendant zu den MS-DOS-Tools und WINI-Tools für den ATARI ST, geschrieben wurden lassen sich tatsächlich übertragen, wenn diese sich nicht zu nah an ATARI-eigene Betriebssystemaufrufe (z.B. LineA, XIOS, BIOS, GEMDOS) halten. Bei konvertierten GFA-BASIC-Programme hatte ich es sogar noch einfacher, da der Konverter Syntax und C-Konvention genau auf die Portierung abrichtet. Kompatibel vorhanden sind alle an GFA-BASIC angelehnten Grafikfunktionen von Linien über Boxen bis zu gefüllten Ellipsen, soweit ich dies feststellen konnte. Befehle wie PUT und GET geben zum Glück, wie auf dem ATARI gewohnt, Strings zurück und können beliebig große Bildausschnitte bearbeiten. Auch bei der Textausgabe entsprechen die Koordinaten im Gegensatz zu „GFA-BASIC PC" den ATARI-Gewohnheiten. Besonders erstaunt hat mich, daß alle gewohnten Füllmuster und Mauszeiger vorhanden waren und funktionierten. Selbst die ATARI-eigenen Füllmuster lassen sich also übertragen. Einen Fehler fand ich jedoch bei der farbigen Ausgabe selbstdefinierter Muster. Während sonst alle Muster mit Farben korrekt auf den PC übertragen wurden, färbte Windows nur den äußeren Rand dieser Muster. Ebenso können nur 16 der 32 Bytes der selbstdefinierten

```
/* GFA-BASIC -> C Übersetzung vom Fri Jul 17 11:20:46 1992
*/
#include <b_nach_c.h>


/* Globale modulinterne Prozeduren/Funktionen */

void    P_fuellen(void);
void    P_mouse(void);
void    P_texte(void);

/* Variablen: */

int     iI;
int     jI;
long    aL;
Bstr    aS;
Bstr    textS;


void    main(void)
{
        /* Initialisierung */
        MO = 0;
        MF = 1;
        DYN_STRINGS = 1;
        CHAR = C_HAR;
        MAXS = M_AXSP;
        SMAX = 22;
        B_init(1);
        B_hidem();

        /* Benutzerprogramm */

        B_drealloc(&aS,C_HAR);
        B_drealloc(&textS,C_HAR);
        P_fuellen();
        P_mouse();
        P_texte();
        B_exit(0);
}

void    P_fuellen(void)
{
        B_cls();
        B_deffill(3,2,6);
        B_pbox( 0,0,(B_WORK_OUT[0]+1-OX),(B_WORK_OUT[1]+1-
                                                      2*OY));
        for(jI=0;jI<=10;(jI)++) {
                B_deffill(jI,3,3);
                B_pbox( 10,10,30,30);
                aS = B_get(10,10+19,30,30+19,aS);
                for(iI=0;iI<=15;(iI)++) {
                        B_put(30+iI*50,10+19+30*jI,aS,iI);
                }
        }
        aL = B_keyget();
        B_cls();
}

void    P_mouse(void)
{
        B_cls();
        B_printf(-1,-1,"Mouse-Zeiger - Bitte Taste
                                              drücken\n");
        for(iI=0;iI<=7;(iI)++) {
B_printf(1,2,"Mouse-Zeiger Nr.: %d  \n",iI);
                B_defmouse(iI);
                aL = B_keyget();
        }
        B_defmouse(0);
        B_cls();
}

void    P_texte(void)
{
        Bstr    testS = B_ldrealloc(C_HAR);
        int     farbeI = 0;
        int     artI = 0;
        int     winkelI = 0;
        int     groesseI = 0;
        textS=B_nsprintf(textS,"Hello BASIC nach C",18);
        for(farbeI=1;farbeI<=15;(farbeI)++) {
                for(groesseI=0;groesseI<=20;(groesseI)++) {
B_deftext(farbeI,artI,winkelI,groesseI,-1);
                        B_text(0,20+20*groesseI,0,textS);
                }
                aL = B_keyget();
                B_cls();
        }
        B_dsfree(testS);
}
```

... und nach der Konvertierung

Füllmuster gesetzt werden. Da dies bei den MS-DOS-Tools korrekt funktioniert, tippe ich auf eine Windows-Eigenart.

Schriften

Im gesamten Grafikbereich konnte ich lediglich bei der Schriftenausgabe der WINI-Tools und bei der Abrundung der Linienenden Inkompatibilitäten bemerken, die laut Herstellerangaben auf die Eigenarten von Windows zurückgehen. Vorteil zu diesem Nachteil: durch die Vielzahl der Windows-Fonts (z.B. True Type) lassen sich die Programme mit ansprechenden Schriften, die neuerdings seit Windows 3.1 auch in allen Größen und Richtungen skalierbar sind, verschönern. Gelungen nach meiner Meinung ist die Möglichkeit, durch ein vorgeschaltetes Menü den Lieblings-Font vom Benutzer wählen zu lassen - dies legt man durch Hinzu-Linken einer besonderen Bibliothek fest. Bei den MS-DOS-Tools ist das Problem der Schriften dadurch gelöst, daß immer mit dem original ATARI-Zeichensatz ausgegeben wird, d.h. alle vom ATARI gewohnten Bilder sind auch identisch auf dem PC. Gestört hat mich hier jedoch, daß die Schriftausgabe beim Bildschirm-Scrollen relativ langsam ist. Bei CICERO hat man mir versprochen, die Ausgabe des Scrollings so weit wie möglich zu optimieren.

Generell kann ich zum Vergleich GFA-BASIC PC mit WINI-Tools und MS-DOS-Tools von CICERO im Grafikbereich feststellen, daß eine 1 zu 1 Programmübertragung, sei es mit konvertierten GFA-BASIC oder selbst geschriebenen C-Programmen, zur Zeit nur mit den Tools von CICERO tatsächlich möglich ist. Zur Demonstration ist ein kleines Grafikbeispiel angefügt.

Jeder ATARIaner, der zum PC überwecheln muß, kennt das typische PC-Problem: die MS-DOS-Speicherverwaltung. Bei den IBM-kompatiblen PCs gibt es selbst bei den heute gebräuchlichen 386ern und 486ern immer noch die Beschränkung des Speichers auf 640 KByte und der auf 64 KByte beschränkten Seitengröße des Speichers - keine Chance also für einen linearen Speicher. Dies bedeutet, daß es nicht möglich ist, größere Felder und Variablen als 64 KByte anzulegen. Für den nicht gerade verwöhnten GFA-BASIC-Programmierer mit den auf 32 KByte beschränkten Strings mag dies zuerst nicht sehr bedeutsam erscheinen. Da jedoch Bilder und ganz besonders VGA-Farbbilder leicht in der Größenordnung von mehreren hundert KByte liegen, ist dies das ‚Aus' für Befehle wie SGET und SPUT. Die gleiche Schwierigkeit hatten schon die GFA-BASIC-Programmierer auf dem TT. Die MS-DOS-Tools, die dieses Problem am härtesten trifft, helfen sich dadurch, daß sie große Bilder auf Festplatte oder EMS auslagern. Die WINI-Tools erlauben jedoch wie die ATARI-Tools die Programmierung von sogenannten ‚Großen Strings', die nicht mehr in der Größe beschränkt sind. In diese Strings können durchaus Bilder von der Größe eines Megabytes eingelagert und wieder auf den Bildschirm gezaubert werden.

Bei diesem Test habe ich aus Platzgründen nur die oben beschriebenen beiden Programmierbereiche darstellen können. Die Möglichkeiten der Tools hier auch nur annähernd vollständig darzustellen, würde den Rahmen sprengen. Insgesamt machen die Tools von CICERO jedoch bisher einen soliden und gut durchdachten Eindruck. Wenn auch nicht alle meine GFA-BASIC- Programme sofort auf dem PC liefen oder ich doch einiges an Eingewöhnungszeit benötigte, muß ich sagen, daß die MS-DOS- und WINI-Tools die Einarbeitung in die Grafikprogrammierung des IBM-kompatiblen PCs vollständig erübrigen, was besonders bei Windows ein nicht zu unterschätzender Vorteil ist. Mit den Tools ist eine enorme Zeitersparnis bei der Portierung von Programmen auf andere Betriebssysteme gegeben. C-Programmierer, die ihre Programme auf der Grundlage der Tools schreiben, können gewiß sein, daß die Portierung mühelos automatisiert abläuft, da alle CICERO-Tools kompatible Bibliotheksfunktionen festlegen. Wenn CICERO das Vorhaben wahrmachen sollte, die Tools auch für UNIX und Apple herauszugeben, wäre dies möglicherweise endlich die Lösung für eine über alle Systeme kompatible Grafikprogrammierung.

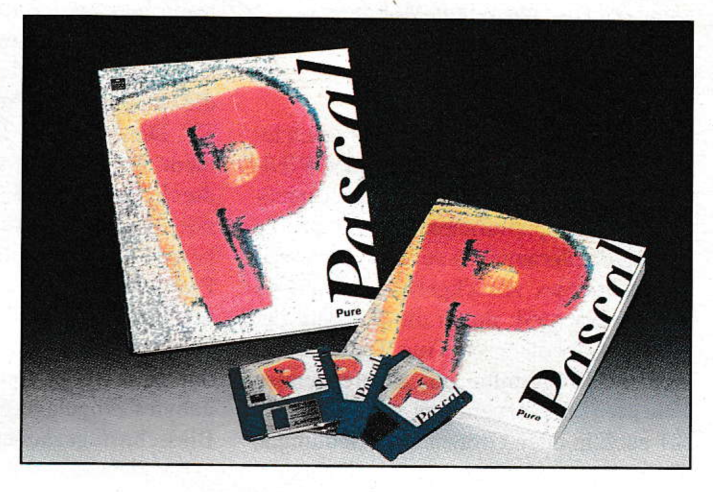
Michael Regitz

Preise:

BASIC nach C MS-Windows	*549,-*
Wini-Tools (einzeln)	*249,-*
MS-Tools (einzeln)	*249,-*

Bezugsquelle:
Cicero Innovative System-Software
Ballweilerstraße 7
6676 Mandelbachtal
Tel.:(06803) 2834

Pure Pascal



Turbo Power für ATARI-Rechner

Pascal hat leider auf ATARI-Rechnern nicht die Verbreitung gefunden, wie es im MS-DOS-Bereich der Fall ist. Von Anfang an spalteten sich die ATARI-Programmierer in BASIC- und C-Spezialisten auf. Nur wenige ließen sich nicht beirren und frönten ihrer Lieblingssprache PASCAL. Mit Maxon PASCAL kam dann frischer Wind in die PASCAL-Szene auf dem ATARI. Dieser Compiler machte den Turbo-Pascal-Standard verfügbar. Jetzt ist Pure Pascal, der jüngste Sproß der PASCAL-Compiler, mit seiner Kompatibilität zu Turbo Pascal 6.0 erhältlich.

Geliefert wird Pure Pascal in einem hübschen farbigen Einschubkarton mit einem ebenso aufgemach ten, 300 Seiten umfassenden gebundenen Handbuch. Auf den drei Disketten befinden sich neben der Entwicklungsumgebung, in die Compiler, Linker und Debugger integriert sind, auch einige Ordner mit Beispielprogrammen zur Programmierung mit Pure Pascal sowie die Bibliotheken und der Assembler.

Systemanforderungen

Für ein angenehmes Arbeiten sollten schon 2 MB RAM und Festplatte vorhanden sein. Mit 1 MB RAM kann zwar gearbeitet werden,jedoch sind die Grenzen des Systems dann bald erreicht. Die Installation ist sehr einfach und beschränkt sich auf das Kopieren der Disketten in ein Verzeichnis der Festplatte. Eine Komplettinstallation beansprucht etwa 1.7 MB. Es wurde Sorge getragen, daß auch ein Betrieb mit Diskettenlaufwerken möglich ist, wobei aber Komforteinbußen in Kauf genommen werden müssen.

Dokumentation

Sie hinterläßt ein etwas zwiespältiges Bild. Die Beschreibungen der Arbeitsoberfläche, des Editors, des Compilers und des Debuggers sind sehr ausführlich und geben eine gute Hilfestellung. Der Stil ist sehr locker und der Text wird auch schon mal mit einem kleinen Scherz bereichert. Gut sind auch ein Crash-Kurs für Kurzeinsteiger und die Pascal-Einführung für Anfänger. Kleine Beispielprogramme geben ein Gefühl für die neue Sprache. Vorbildlich ist auch eine Einführung in die GEM-Programmierung, bei der neben der Programmierung von Fenstern und Dialogen auch gleich eine Unit zur Initialisierung eines GEM-Programmes und der wahlweise Start als ACC mitgeliefert wird. Unerläßlich, weil hier doch ein gewisses Neuland beschritten wird, ist ein Kapitel über OOP (objekt orientierte Programmierung). Im Handbuch werden knapp, aber ausreichend die Grundlagen dargestellt. Angaben weiterführender Literatur am Ende der Kapitel bieten die Gelegenheit, sich intensiver als in einem Handbuch mit der Materie zu befassen. Neben diesen positiven Seiten möchte ich auch Kritik äußern. So sind die Beschreibungen von Nutzung und Bedienung, sowie des Einbindens von Routinen des Assemblers entschieden zu kurz ausgefallen. Es gibt keine Kurzzusammenfassung des Leistungsumfanges und der Einschränkungen von Pure Pascal. Der Index ist mit 12 Seiten völlig unterdimensioniert. Fehlermeldungen können nicht direkt im Handbuch nachgeschlagen werden, da der Index nur global angibt, wo diese zu finden sind, und daher alle Fehlermeldungen, bis man fündig geworden ist, durchsucht werden müssen. Es gibt eine Kurzübersicht der Units, leider ebenfalls ohne Index, mit einer Kurzbeschreibung der Befehle. Eine Kurzbeschreibung mit den zu übergebenden Parametern fehlt. Um an diese zu gelangen, muß die Online-Hilfe bemüht werden.

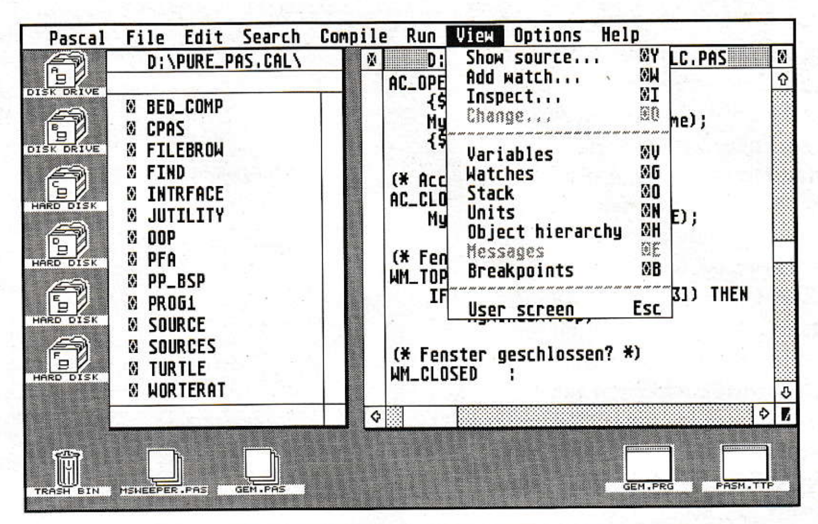
Oberfläche

Nach dem Start von Pure Pascal erscheint eine GEM-Desktop-ähnliche Oberfläche, die anfangs etwas gewöhnungsbedürftig ist, dank der komfortablen Bedienung aber nach kurzer Zeit ein besonders angenehmes Arbeiten ermöglicht. Dateien und Programme können als Icons auf die Oberfläche gezogen werden. Neben den üblichen Funktionen bietet das File-Menü auch die Anlage neuer Ordner und die Formatierung von Disketten an. Dies erspart ein öfteres Verlassen von Pure Pascal. Der Editor ermöglicht ein Einrücken des Sourcecodes mit „dehnbaren Tabulatoren". Je nach eingestellter Tabulatorenweite wird der Text automatisch eingerückt. Bei Änderungen der Tabulatorenweite ist der Text gleich entsprechend neu formatiert. Der Editor benutzt hierfür keine Space-Zeichen, sondern ein anderes Steuerzeichen, welches bei manchen anderen Editoren etwas unschön aussehen kann. Die sehr

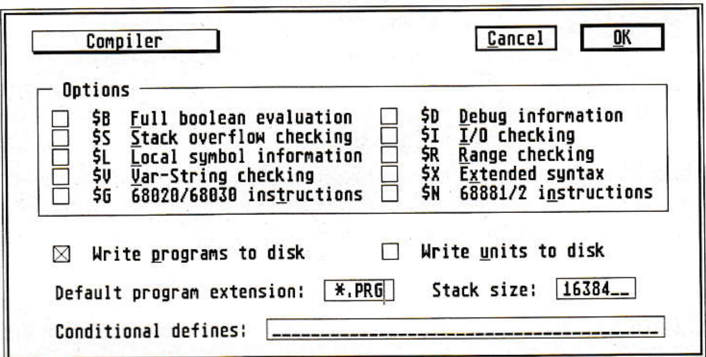
umfangreichen Such- und Ersetzfunktionen lassen keine Wünsche mehr offen. Gut ist auch die „Live Scroll-Funktion". Mit ihr erfolgt das Scrolling der Fenster, schon während der Slider bewegt wird. Hierdurch bleibt die alte Slider-Position sichtbar und kann wieder angefahren werden, um an gleicher Stelle fortfahren zu können. In weiteren Menus können die Compiler-Aufrufe usw. durchgeführt und die entsprechenden Einstellungen vorgenommen werden. Auch hier hat das Konzept einige Nachteile. Wegen des Debuggers handelt es sich bei der Pure-Pascal-Oberfläche nicht um ein GEM-Programm. ACCs lassen sich daher erst nach Selektion eines Menüeintrages ausführen. Ein weiterer Nachteil ist darin zu sehen, daß die Oberfläche nicht zukunftssicher programmiert ist. Zwar ist Pure Pascal auflösungsunabhängig betreibbar, arbeitet aber nicht mit dem in Bälde zu erwartenden Multi-TOS zusammen. Es kommt dabei zu dem Effekt, daß die Menüs von Desktop und Pure Pascal übereinander ausgegeben werden und damit ein Arbeiten unmöglich ist. Dazu werden auch noch ohne Verwendung von XBRA einige Vektoren verbogen. Hier sollten dringend einige Änderungen erfolgen.

Compiler

Der Compiler ist über viele Optionen, die alle aufzuzählen den Rahmen dieses Tests sprengen würde, steuerbar. Neben der Möglichkeit, z.B. Debug-Informationen und Funktionsaufrufe ohne Dummy-Rückgaben einzustellen, ist für Besitzer eines TTs oder einer PAK die Möglichkeit zur Ausnutzung der 68020/30-Prozessoren und der mathematischen Coprozessoren bestimmt erfreulich. Der Compiler nimmt seine Arbeit sehr gründlich wahr und deckt manchmal Schwachstellen auf, die andere Compiler nicht wahrnehmen. Dabei sind die Übersetzungszeiten sehr klein. Auch bei größeren Projekten bleibt eine Kaffeepause erspart. Bleibt die Frage nach dem erzeugten Code. Wie aus den Tests ersichtlich ist, sind kleine Programme bei Pure Pascal etwas größer dimensioniert als üblich. Dies fällt aber bei größeren Projekten nicht ins Gewicht. Hier kann Pure Pascal mit einem sehr kompakten Code glänzen. Dieser macht sich auch in der Ausführungsgeschwindigkeit bemerkbar. Der Compiler ermöglicht auch die bedingte Compilierung. Damit können Programme an bestimmte Begebenheiten angepaßt werden (wie z.B. ein vorhandener Coprozessor), ohne daß dafür zwei verschiedene Programme geschrieben werden müssen.



Die Oberfläche von Pure Pascal



Die Compiler-Optionen

Eine Optimierung z.B. von Schleifen wird vom Compiler selbständig durchgeführt. Treten Fehlermeldungen auf, bricht Pure Pascal nicht bereits beim ersten Fehler ab, sondern compiliert das ganze Programm. Die Fehlermeldungen erscheinen in einem Fenster, und nach Anwahl springt der Editor an die entsprechende Stelle. Bei Unklarheiten hilft die Online-Hilfe weiter. Sie ist sehr ausführlich gehalten, und es gibt meist zur Verdeutlichung noch ein lauffähiges Kurzprogramm. Neben den üblichen Fehlermeldungen liefert der Compiler zusätzlich noch Warnungen zurück. Diese führen nicht zu einem Abbruch, und auch die erzeugten Programme laufen ohne Fehler. Warnungen sollten eher als Hinweis verstanden werden z.B. für Variablen, die nicht benutzt wurden, oder Funktionen, deren Rückmeldungen nicht ausgewertet wurden. Für größere Projekte sind diese Warnungen sehr nützlich, da hier oft „Variablenmüll" auftreten kann, der sich so vermeiden läßt. Aufgefallen ist mir noch ein kleiner Fehler in der Online-Hilfe. Im Handbuch wird ein Beispiel gebracht, mittels des Befehls PTR einem Zeiger eine Adresse zuzuweisen. In der Online-Hilfe steht dazu, daß dieser Befehl nur für 80X86 Prozessoren sinnvoll sei. Seltsamerweise ist der Befehl aber implementiert und arbeitet auch, wie er soll. Bei Größe und Umfang der Online-Hilfe lassen sich kleine Fehler wohl kaum vermeiden, und es kann davon ausgegangen werden, daß sie bald behoben sind.

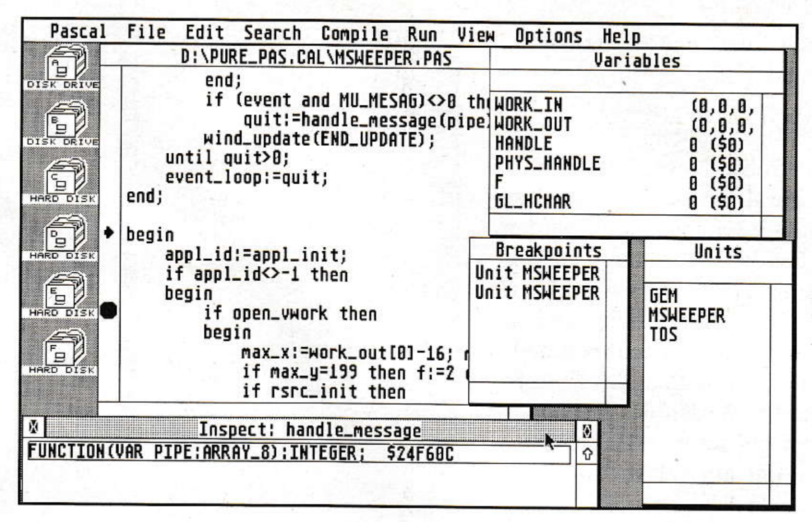
Assembler

Er liegt im System als TTP vor und bietet ein weites Anwendungsspektrum. Es werden alle 680X0-Prozessoren unterstützt, ebenso die mathematischen Coprozessoren. Dank der vielen Optionen und der umfangreichen Fehlermeldungen, wozu auch eine eigene Online-Hilfe gehört, ist eine effiziente Einbindung von Assemblersourcen kein Problem. Die Darstellung aller Optionen und Möglichkeiten würde den Rahmen dieses Artikels übersteigen. Da Pure Pascal schon sehr schnell ist, wird sich der Einsatz wohl meist nur auf besonders zeitkritische Problemstellungen beschränken.

Debugger

Für die Fehlersuche unerläßlich und in den bekannten Pascal-Compilern nicht vorhanden ist der Debugger, der immer dann weiterhilft, wenn keine syntaktischen Fehler vorliegen. Bei Pure Pascal wird der Debugger nicht extern gestartet, sondern ist im System integriert. Er arbeitet auf Quelltextebene, zeigt also den gerade abgearbeiteten Quelltext an. Ein kleiner Pfeil an der linken Seite zeigt an, wo sich gerade der Programm-Counter befindet. Daneben kann der Wert von Variablen beobachtet, aber auch verändert werden. Die Ausgabe erfolgt über Fenster. Weitere Fenster für Stack, Units, Breakpoints und Objekthierarchie lassen sich öffnen. Kom-

fortabel ist auch das Unit-Fenster. Durch Doppelklick erhält man ein weiteres Fenster, in dem die Konstanten, Typen, Prozeduren, Funktionen und globalen Variablen der Unit angezeigt werden. Durch direkte Tastatureingaben kann ohne langes Scrollen eine interessierende Stelle angefahren werden. Durch den Einbau von Breakpoints lassen sich Programme an beliebiger Stelle unterbrechen. Mit den bestehenden Möglichkeiten dürften sich wohl praktisch alle Probleme lösen lassen. Die Vielzahl der Möglichkeiten führt zu einem gewissen Problem, die vielen Fenster auf dem Bildschirm unterzubringen. Eine höhere Auflösung ist deshalb sehr empfehlenswert.



Der Debugger

Bibliotheken

Ohne sie wäre der Pascal-Compiler wie ein Segler ohne Segel. In diesem Falle kommt noch hinzu, daß Pure Pascal sowohl den PC- als auch den ATARI-Bereich abdecken muß, ohne daß es zu Inkompatibilitäten kommen darf. Im Lieferumfang befinden sich zur Zeit sechs Bibliotheken. Den DOS-konformen Bildschirm-I/O übernimmt die CRT-Unit. Sie enthält Routinen für die Einstellung verschiedener Textmodi und -farben. Dateioperationen können ebenso auf den Bildschirm umgeleitet werden, wie sich Fenster für die Textausgabe erzeugen lassen. Die DOS-Unit stellt die dem GEMDOS ähnlichen Befehle bereit. Sehr wichtig für die Kompatibilität zu Turbo Pascal sind Routinen zur Interrupt-Programmierung (GetIntVEc und SetIntVec), zur Programmausführung (DosExitCode, Exec, Keep und Scrap Vectors), sowie Routinen für die Environment-Variablen und DOS-Parameter. Unerläßlich ist für die Grafikausgabe die Graf-Unit. Auch hier hat man sich sehr bemüht, so nah wie möglich an das Vorbild heranzukommen. Sogar BGI-Treiber und Vektorzeichensätze lassen sich installieren. Ein VDI.BGI-Treiber liegt bei. In der System-Unit sind die Standard-Pascal-Befehle zusammengefaßt worden. Neben dem DOS-Bereich ist natürlich die Einbindung des ATARI-Bereiches von großer Wichtigkeit. Die unteren Bereiche des Betriebssystems werden mit der TOS-Unit abgedeckt. Diese enthält die bekannten Routinen von BIOS, XBIOS und GEMDOS. Die Befehle der neuen Betriebsystemversionen sind bereits integriert. Der GEM-Unit wurden noch einige nützliche Routinen beigefügt, die das Programmieren sehr erleichtern. Neben den bekannten AES- und VDI-Routinen sind noch einige Auskunfts- und Eingaberoutinen für die Ermittlung bzw. das Setzen von Pascalstrings in/aus TEDINFO-Strukturen, Icons usw. hinzugekommen. Leider fehlen die Routinen für die Programmierung von CPX-Modulen. Im Handbuch ist eine Auflistung der FSMGDOS-Befehle enthalten. Seltsamerweise sind diese Befehle nicht mit der Online-Hilfe abrufbar. In Ermangelung von FSMGDOS konnte ich leider nicht ermitteln, ob die Befehle bereits implementiert sind.

Kompatibilität

Wie bereits in den Bibliotheksbeschreibungen ersichtlich, hat man bei Application Systems viel Wert auf eine bestmögliche Anpassung an Turbo Pascal 6.0 gelegt. Diese Bemühungen tragen in der Praxis ihre Früchte, machen sie doch die Anpassung von PC-Programmen an den ATARI sehr einfach. Die meisten Programme, die ich ausprobierte, ließen sich ohne große Schwierigkeiten direkt compilieren und ausführen. Oft mußten nur die benutzten

Sieb des Erasthotenes :	Primzahlenberechnung
Fibonacci :	Rekursionstest
Float :	Fließkommaarithmetik
Quicksort :	Rekursionen und Entscheidungen
Savage :	Genauigkeit von Fließkommaoperationen
Drhystone :	Datenflußanalyse, Stringcopy
Kurzprogramm :	Länge eines leeren Programms
Referenzrechner:	Mega ST 2, 16 MHz-Speeder, TOS 1.04

Die erste Spalte gibt jeweils die Zeiten in Sekunden eines 8-MHz-Rechners wieder, die zweite Spalte den gleichen Durchlauf unter Verwendung des 16-MHz-Speeders. Niedrige Zeiten bezeichnen bessere Werte mit Ausnahme der Drhystone-Benchmarks, wo höhere Werte für linear bessere Ergebnisse stehen. Bei allen REAL-Operationen wurde mit maximaler Genauigkeit gearbeitet. Je kleiner der Wert der Genauigkeit von Null abweicht, desto genauer wird berechnet.

Benchmarks	Maxon-Pascal	Pure-Pascal	ST-Pascal	Maxon-Pascal	Pure-Pascal	ST-Pascal
Erastothenes	2,50	0,78	2,59	1,37	0,43	1,39
Fibonacci	66,91	18,66	40,64	41,94	10,92	24,38
Float	62,63	45,44	58,31	33,32	23,83	29,37
Quicksort	96,30	65,85	94,55	50,65	35,84	53,05
Savage	80,32	72,62	158,99	40,59	37,97	87,35
Drhystone	645,16	1700,68	915,75	1127,40	2989,54	1549,19
Genauigk.	-2,2686585E-09	-2,1719515E-11	-5,2684436E-01	-2,2686585E-09	-2,1719515E-11	-5,2684436E-01
Kurzprog.	972Bytes	3072Bytes	334Bytes			

Unit-Namen etwas angepaßt werden. Aufwendiger wird es erst dann, wenn sich die PC-Programmierer zu sehr mit der rechnerspezifischen Hardware befassen oder Assembler-Routinen einbinden. Hier ist es nur verständlich, daß Pure Pascal an dieser Hürde scheitern muß. Es bleibt aber festzuhalten, daß Pure Pascal der kompatibelste Turbo Pascal-Compiler auf dem ST ist.

Geschwindigkeit

Um hierüber Aussagen machen zu können, habe ich einige Tests durchgeführt und Pure Pascal mit den direkten Konkurrenten verglichen. Wie aus der Tabelle entnommen werden kann, bleibt Pure Pascal in jedem Test der Sieger. Damit erzeugt Pure Pascal den zur Zeit schnellsten Code aus Pascal-Sourcen. Die Drhystone-Werte erreichen das Niveau eines C-Compilers (Pure C). Wer also immer der Meinung war, Pascal-Programme seien langsam, wird von den Ausführungszeiten überrascht sein. Wenn berücksichtigt wird, daß ein Apple II für einen Durchlauf FIB(20) etwa eine Minute brauchte, zeigt dies, wie gewaltig die Rechenleistung zwischenzeitlich angestiegen ist.

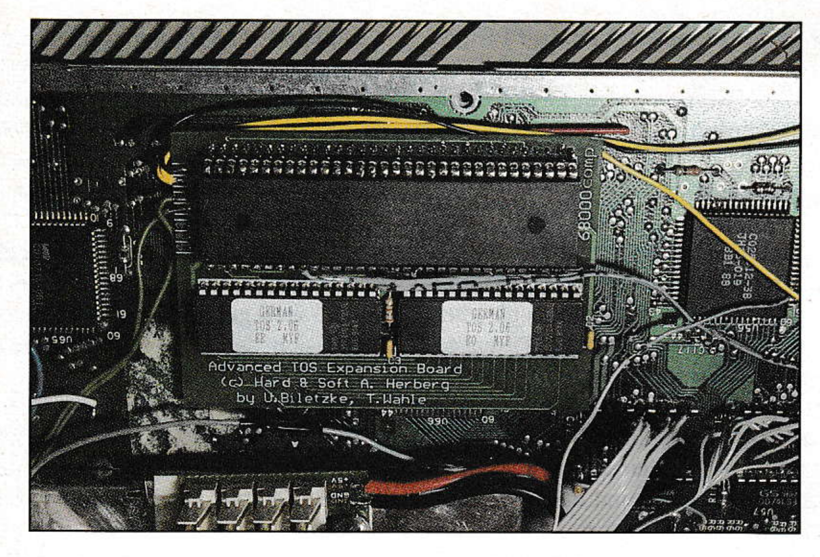
Resümee

Das Warten auf Pure Pascal hat sich gelohnt. Der Anwender erhält nicht nur einen sehr komfortablen, sondern gleichermaßen schnellen und zu Turbo Pascal 6.0 kompatiblen Compiler, der über einen ausgezeichneten Debugger und die Möglichkeit zur objektorientierten Programmierung verfügt. Die kleinen Ungenauigkeiten der Online-Hilfe und das „Nichtvorhandensein“ bestimmter Befehle werden wohl bereits im nächsten Update ausgebügelt sein. Schwerer wiegt da schon die die Entwicklung hemmende nicht GEM-konforme Oberfläche, die hoffentlich in bezug auf Multi-TOS überarbeitet wird. Eine weitere Kritik möchte ich auch am Gesamtumfang üben. Irgendwie habe ich immer das Gefühl, die Hersteller seien der Meinung, Pascal-Anwender bräuchten weniger Information und weniger Zusatzprogramme als z. B. C-Programmierer. Ein Vergleich ist in diesem Falle möglich, da ASH mit Pure C ein Schwesterprodukt anbietet, das genausoviel kostet wie Pure Pascal. Während das Pascal-Handbuch 300 Seiten umfaßt, sind es bei Pure C 650 Seiten. Dazu gibt es bei Pure C noch ein Resource Construction Set, welches Pure Pascal auch gut zu Gesicht stünde. Hier sollte sich der Hersteller eine gewisse Angleichung überlegen.

Dies soll aber nicht die Leistungen, die in Pure Pascal stecken, schmälern. Jeder, der in Pascal programmieren möchte und auch den Zugriff auf die riesigen Mengen an Bibliotheken und Sourcen aus dem DOS-Bereich haben möchte, ohne auf eine hohe Geschwindigkeit verzichten zu wollen, wird kaum an Pure Pascal vorbeikommen.

Rainer Esser

Bezugsquelle:
Application Systems Heidelberg
Englerstraße 3
W-6900 Heidelberg 1
Tel.:(06221) 300002



TOS 2.06 neu aufgelegt

In der Märzausgabe dieses Jahres berichteten wir über die TOS 2.06 Extension Card der Firma Artifex Computer GmbH. Mittlerweile gibt es ein solches TOS-Update auch von anderen Herstellern, und Artifex selbst hat ihre Hardware überarbeitet. Was dabei herausgekommen ist, zeigt dieser Bericht.

Nach langem hin und her hat Artifex Anfang dieses Jahres ein TOS-2.06-Update (auch TEC genannt) nebst erforderlicher Hardware auf den Markt gebracht. Die damals noch recht große Platine war für den Einbau in einige STs doch recht hinderlich, ganz zu schweigen von den vielen Kabeln, die verlötet werden mußten. Das bewog die Firma Hard & Soft eine eigene, kompaktere Karte zu entwickeln, wobei Artifex die beschriebenen Nachteile wohl auch erkannt und die Hardware der TOS Extension Card einer gründlichen Überarbeitung unterzogen hat.

Wir haben also die TOS 2.06-Erweiterungen beider Hersteller unter die Lupe genommen und, um es gleich vorwegzunehmen, bei beiden keine gravierenden Mängel feststellen können. Aber bei einigen Kleinigkeiten gibt es doch Unterschiede.

Der Einbau ...

... stellt sich bei beiden Produkten als sehr einfach heraus. Die TEC (TOS Extension Card) wird auf den Prozessor gesteckt, wozu man ggf. noch einen 64poligen Sokkel auf den 68000er löten muß. Ist die CPU bereits gesockelt, kann die TEC auch in diesen und der Prozessor auf die Erweiterungskarte gesteckt werden. Die TOS 2.06 Karte von Hard & Soft ermöglicht schon vom Platinen-Layout her eine Umschaltung zwischen neuer und alter TOS-Version, wobei Artifex diese Umschaltmöglichkeit zwar auch im Handbuch beschreibt, allerdings nicht auf ihrer Platine vorgesehen hat. Somit ist hier etwas aufwendigere Lötarbeit erforderlich.

Für Besitzer eines MegaSTs bietet Artifex eine TEC für den Megabus an - damit entfällt dann auch der Turmbau auf der CPU, und der Einbau ist in zirka fünf Minuten vorgenommen.

Die Software

Beiden Erweiterungen liegt eine Diskette mit Software bei: Ataris Kontrollfeld samt den Standard-CPX-Modulen. Desweiteren sind auf der Diskette, die Artifex mitliefert, ein Programm zur Stepratenumschaltung, zur Verkürzung der Boot-Zeit nach einem Kaltstart und einige neue Icons enthalten.

Das Handbuch von Artifex erläutert den Einbau der TOS Extension Card (sowohl CPU- als auch Megabus-Version) recht ausführlich und beschreibt natürlich die neuen Funktionen des TOS 2.06. Das Handbuch von Hard & Soft enthält prinzipiell die gleichen Informationen, geht aber auf die Funktionen des neuen TOS ausführlicher ein.

Fazit

Wir haben bewußt auf eine Funktionsbeschreibung des TOS 2.06 verzichtet, weil wir sie bereits in der Märzausgabe behandelt haben, und uns somit auf hardwaremäßige Realisierung der TECs beschränkt.

An dieser Stelle erwartet man meist eine Kaufempfehlung, die wir hier jedoch nicht geben können. Will man auf beide TOS-Versionen umschaltbar zugreifen können, sind bei der Artifex-Karte zirka drei Verbindungen mehr zu löten als bei der von Hard & Soft. Dafür ist die TEC von Hard & Soft ein wenig größer als die der Firma Artifex. Es sind also lediglich Feinheiten, die die beiden Karten noch voneinander unterscheiden.

Robert Osten

Die Preise:

TOS Card 2.06 (Hard & Soft): 149,- DM

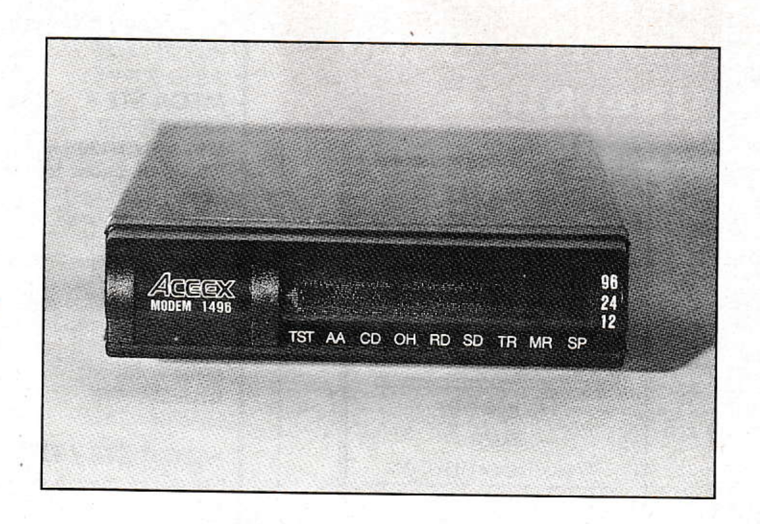
TOS Extension Card (Artifex)
(CPU- und Megabus-Version): 149,- DM

Bezugsadressen:

Hard & Soft Computerzubehör GmbH
Obere Münsterstraße 33-35
W-4620 Castrop-Rauxel
Tel.:(02305) 18014

Artifex Computer GmbH
Holbeinstraße 60
W-6000 Frankfurt 70
Tel.:(069) 6312456

Low-Cost - High-Speed-Modem



Mitte letzten Jahres stellten wir Ihnen fünf High-Speed-Modems vor. Mittlerweile hat sich jedoch, wie der interesierte Leser bestimmt schon bemerkt hat, einiges auf dem Markt getan: Geräte mit DSP liegen nun unterhalb der 1000 DM-Grenze, und es gibt einige Low-Cost-Prod kte, die noch günstiger sind und anscheinend das gleiche wie die etwas teureren Brüder leisten. Wir haben uns zwei Angebote herausgepickt...

Diese zwei Modems waren schon gleich nach dem Öffnen der Verpackung für eine Überraschung gut: Sie sind baugleich und haben lediglich eine unterschiedliche Typenbezeichnung. Das Gerät der Firma TKR GmbH in Kiel nennt sich IM-144VF+, wobei IM andeutet, daß es sich um ein Exportgerät handelt. Das zweite Modem kommt vom DTP Service Julian Riedlbauer, ein in ATARI-Kreisen wohl eher unbekannteres Unternehmen. Dort heißt das gleiche Modem ACEEX DM 1496, was der Original-Bezeichnung des taiwanesischen Herstellers wohl am nächsten kommt: ACEEX 1496.

Einschalten

Hat man das Modem verkabelt und schaltet es ein, gibt der eingebaute Lautsprecher zur Begrüssung ein ‚knack' von sich. Mit dem (Standard-)Befehl L0 bis L2 läßt sich die Lautstärke des Lautsprechers einstellen, aber leider kann der Anwender nur zwischen ziemlich laut, sehr laut und unerträglich laut wählen. Selbst die geringste Einstellung ist für einen Wohnraum mittlerer Größe unangenehm - also haben wir uns mit dem Einbau (Vorsicht: Garantieverlust!) eines kleinen Widerstandes beholfen (übrigens 82 Ω), der die Lautstärke auf ein erträgliches Maß senkt.

An der Frontseite sind acht rote LEDs angebracht, die über den momentanen Zustand des Gerätes Auskunft geben. Rechts außen finden sich noch drei weitere LEDs zur Geschwindigkeitsanzeige. Offline geben Sie die Geschwindigkeit der seriellen Schnittstelle an, online die der aufgebauten Verbindung.

Konfiguration

Wie die Modems nun auch immer genannt werden, sie kommen vom gleichen Hersteller und sind somit völlig identisch in der Funktion. In Bild 1 sehen Sie die Aufmachung der Modemkonfiguration, die an das Terminal gesendet wird, wenn sie über AT&V abgerufen wird. Wie zu erkennen, verfügt das Modem über die Möglichkeit, zwei unterschiedliche Konfigurationen dauerhaft zu speichern. Das ist zwar nichts Besonderes, aber durchaus nicht Standard. Leider lassen sich nicht alle angezeigten Konfigurationsmöglichkeiten im NV-RAM (nichtflüchtiger Speicher) speichern. Dazu gehören u.a. auch die Statusregister S7, S9 und S10, die eine wichtige Rolle beim Aufbau einer Verbindung spielen. Die Default-Einstellung dieser Register machte bei uns zwar keine Probleme; sollte jedoch die Zeitspanne zwischen der letzten Ziffer einer gewählten Boxnummer und Beginn eines Carriers sehr lang sein, kann die in S7 spezifizierte Zeitspanne überschritten werden, und das Modem legt auf (wohlgemerkt: Die Werte von S7 usw. lassen sich natürlich verändern, nur halt nicht dauerhaft abspeichern).

Übersichtlich ist die Präsentation der einzelnen Modemeinstellungen nicht gerade, aber es läßt sich damit leben. Wer allerdings schon einmal längere Zeit mit einem Modem von US Robotics gearbeitet hat, wird etwas vermissen...

Betrieb

Kommen wir zu den technischen Daten: Das ACEEX 1496 ermöglicht eine Datenübertragung mit maximal 14400bps (CCITT-Norm V.32bis) und einer Fehlerkorrektur nach V.42/MNP4 bzw. einer Datenkompression nach V.42bis/MNP5. Darüber hinaus beherrscht das Modem den FAX-Standard nach Gruppe3 (9600bps) - sowohl Senden als auch Empfangen und den neuen Standard Class 2. TKR liefert das FAX-Programm Tele Office bzw. Junior Office auf Wunsch gleich mit, natürlich mit Preisnachlaß. Die Firma Riedlbauer dagegen legt dem Modem, ohne Aufpreis, ein FAX-Programm nach Wahl für MS-DOS bei.

Anfänglich gab es im Betrieb noch einige Schwierigkeiten: das Modem baute eine Verbindung von 14400bps auf, konnte sich mit der Gegenstelle jedoch auf kein gemeinsames Protokoll einigen und legte einfach wieder auf. Des weiteren funktionierte nach einem Fallback (z.B. von 14400bps auf 9600bps) das RTS/CTS-Handshaking nicht mehr. Diese Bugs und noch einige mehr (übrigens auch im FAX-Modus) wurden durch eine neue Firmware (im Prinzip das Betriebssystem des Modems) beseitigt, so daß das Gerät mittlerweile recht stabil arbeitet. Selbst beim

Dauerbetrieb an einer Mailbox traten keine gravierenden Probleme auf, natürlich abgesehen von den obligatorischen Anlaufschwierigkeiten.

Wir testeten die Funktionstüchtigkeit mit USR Courier/Dual Standard, Zyxel, Trailblazer (bieten das PEP-Protokoll, ein Multi-Carrier-Verfahren) und Worldblazer (beherrschen außerdem noch V.32bis, also 14400bps, und Turbo-PEP) und, natürlich, mit einem Gerät gleichen Typs (passenderweise hatten wir ja zwei...). Bei allen Tests funktionierte die Datenübertragung zufriedenstellend, sofern wir nicht gerade eine sehr schlechte Telefonleitung erwischt hatten. In solchen Fällen sinkt natürlich die Datenübertragungsrate, da die eingebaute Fehlerkorrektur (meist V.42) defekte Datenblöcke erneut überträgt. Im übrigen sollte man sowieso nach Möglichkeit V.42bis benutzen, da dieses Protokoll zum einen die angesprochene Fehlerkorrektur und auch eine Datenkompression bietet, sofern die Kompression den Datendurchsatz erhöht. Ansonsten werden die Daten einfach ungepackt übertragen (MNP5 dagegen packt egal was kommt, mit dem Ergebnis, daß das Endprodukt unter Umständen länger als die ursprüngliche Datei ist. Folge: die Datenrate geht in den Keller).

Mit dem sehr gebräuchlichen Übertragungsprotokoll ZMODEM und einer V.32bis/V.42bis-Verbindung erreicht man bei einer bereits komprimierten Datei zirka 1650cps (Zeichen pro Sekunde), bei 9600 Bit/s zwischen 1030 und 1040cps. Diese Werte sind zwar keine Rekorde, aber dennoch alltäglicher Durchschnitt weitaus teurerer Geräte.

Zyxel = USR = ACEEX ?

Im Moment dominieren drei Modemtypen auf dem Markt: Das Dual Standard bzw. Courier von US Robotics über 1000,- DM, das Zyxel für knapp unter 1000,- DM und einige andere Geräte für 600,- bis 700,- DM. Unter letzteres fällt zusammen neben Supra und Best auch das ACEEX-Modem. Bis vor kurzer Zeit unterschieden sich diese drei Gruppen von der technischen Leistungsfähigkeit her kaum - bis auf einige Geräte von USR, die noch den HST-Modus bieten (firmeneigenes, asynchrones Protokoll).

Mittlerweile kristallisieren sich jedoch die Unterschiede heraus: Die Low-Cost-Modems arbeiten mit speziellen Chip-Sätzen von Rockwell oder AT&T. Dort ist die Leistungsfähigkeit der Hardware mit 14400bps vorerst ausgeschöpft, eine Erhöhung der Übertragungsgeschwindigkeit ließe sich nur mit anderen Chip-Sätzen und neuer Firmware erreichen. Allerdings ist es dem Anwender kaum zuzumuten, die Chipsätze auszutauschen, wobei man in der Praxis wohl eher ein neues Modem baut als irgendwelche kostenaufwendigen Hardware-Updates anzubieten. Zu diesem Zeitpunkt ist der Gedankengang sowieso hinfällig, da keine leistungsfähigeren Chip-Sätze von Rockwell erhältlich sind (ob von AT&T, entzieht sich unserer Kenntnis).

Das Zyxel arbeitet mit einem DSP (Digitaler Signalprozessor, auch in CD-Playern zu finden) und einem mit 10 MHz getakteten 68000er. Auch die USR-Geräte bieten hardwaremäßig noch einige Reserven, so daß für das Zyxel kürzlich eine neue Firmware erschienen ist, die eine Datenübertragung mit 16.800bps ermöglicht - nur durch das Auswechseln zweier EPROMs, wobei diese Option nur kompatibel zu sich selbst ist. Für die USR-Modems soll es bald etwas ähnliches geben, wobei man auch auf das noch ungenormte V.fast (angeblich 28.800bps) ein Auge geworfen hat. Allerdings wird bis dahin noch einige Zeit vergehen.

Dies ist der eigentliche, große Unterschied zwischen den unterschiedlichen Geräten: Das ACEEX 1496 ist mit FAX G3 und V.32bis hardwaremäßig ausgelastet, eine Erweiterung auf spätere und schnellere Normen ist nicht ohne weiteres möglich.

```
RUFUS  Datei  Fenster  Programm  Parameter                          Mit 22/07
RUFUS v1.11 rel.9
at&v
ACTIVE PROFILE:
B0 E1 L1 M2 Q0 V1 W1 X3 Y0 &C1 &D2 &G0 &J0 &L0 &P1 &Q5 &R0 &S0 &X0 &Y0
%A013 %C0 %P0 %S0 \A3 \C2 \E0 \G1 \H0 \J0 \K5 \N3 \Q3 \T000 \V1 \X1
S00:000 S06:002 S07:060 S08:002 S09:006 S10:014 S11:095 S12:050 S14:8BH
S16:00H S18:00H S21:28H S22:36H S23:08H S25:05H S26:001 S27:CDH
V.42 S REGISTER:S36:007 S46:138 S48:007 S82:128 S95:044

STORED PROFILE 0:
B0 E1 L1 M3 Q0 V1 W1 X3 Y0 &C1 &D2 &G0 &J0 &L0 &P1 &Q5 &R0 &S0 &X0
%A013 %C0 %P0 %S0 \A3 \C2 \E0 \G1 \H0 \J0 \K5 \N3 \Q3 \T000 \V1 \X1
S00:000 S14:ABH S18:00H S21:28H S22:3EH S23:08H S25:05H S26:01H S27:CDH
V.42 S REGISTER:S36:007 S46:138 S48:007 S82:128 S95:044
STORED PROFILE 1:
B0 E1 L1 M0 Q0 V1 W1 X3 Y0 &C1 &D2 &G0 &J0 &L0 &P1 &Q5 &R0 &S0 &X0
%A013 %C0 %P0 %S0 \A3 \C2 \E0 \G1 \H0 \J0 \K5 \N3 \Q3 \T000 \V1 \X1
S00:000 S14:ABH S18:00H S21:28H S22:32H S23:08H S25:05H S26:01H S27:CDH
V.42 S REGISTER:S36:007 S46:138 S48:007 S82:128 S95:044
TELEPHONE NUMBERS:
&Z0=
&Z1=000000000000000000000000000000000000
&Z2=000000000000000000000000000000000000

OK

Keypad=NUM  00:00:00 01:45 001% Modem 1   19200 ZMODEM   VT100 W2
```

Die Konfiguration des Modems läßt sich mit jedem Terminal-Programm (hier: Rufus) erfahren.

Fazit

Das TKR IM-144VF+ kostet 748,- DM, Tele Office im Paket mit dem Modem 138,- DM, und bei Junior Office kommen noch 60,- DM hinzu. DTP Service Riedlbauer bietet das gleiche Gerät unter anderer Typenbezeichnung inklusive MS-DOS-FAX-Programm und zwei Jahren Garantie für 799,- DM an. Auch betreiben beide Firmen eine Service-Mailbox, in der die neuste Firmware für jedermann frei erhältlich ist. Leider sind beide Firmen (die letzte Gemeinsamkeit) nicht die schnellsten, wenn es um die aktuelle Version der Firmware geht, aber uns wurde eine Verbesserung zugesagt.

Der Leser wird an diesem Punkt des Artikel wahrscheinlich eine Kaufempfehlung erwarten, die wir aber nicht liefern können: Mittlerweile werden die Zyxels in Mailboxkreisen (SysOp-Preise oder Sammelbestellungen) auch schon für 800 DM gehandelt, so daß der preisliche Abstand zwischen den beiden Modems immer geringer wird. Dafür ist das ACEEX etwas kleiner, flacher und leiser beim Wählen (Pulswahl). Das Zyxel hingegen ist (noch?) etwas weiter verbreitet und hat auch einige Reserven zu bieten.

Für welches Modem man sich auch letztlich zu entscheiden vermag, es ist in erster Linie eine preisliche Diskussion. Das Datenaufkommen vor allem in den privaten Mailbox-Netzen steigt zwar immer mehr und kann auch das Bedürfnis nach immer höheren Geschwindigkeiten rechtfertigen, aber 14400bps oder, deutlicher, 1650 Zeichen in der Sekunde sind schon recht schnell. Wenn man bedenkt, daß man vor ein bis zwei Jahren für ein 2400-Baud-Modem schon 500,- DM ausgeben konnte, ist fast so etwas wie eine Inflation in diesem Marktbereich eingetreten. Wenn schon ein sehr schnelles, solides Modem benötigt wird, sollte man sich nach Worldblazern (Hersteller: Telebit) oder zukünftig nach V.fast umsehen. Allerdings liegen sowohl US Robotics- als auch Telebit-Geräte auf preislich höherem Niveau (für Privatpersonen i.d.R. unattraktiv), so daß eigentlich nur das Zyxel als Konkurrenz zu dem ACEEX 1496 angesehen werden kann.

Robert Osten

Bezugsadressen:

TKR GmbH
Stadtparkweg 2
W-2300 Kiel 1
Tel.: (0431) 337881

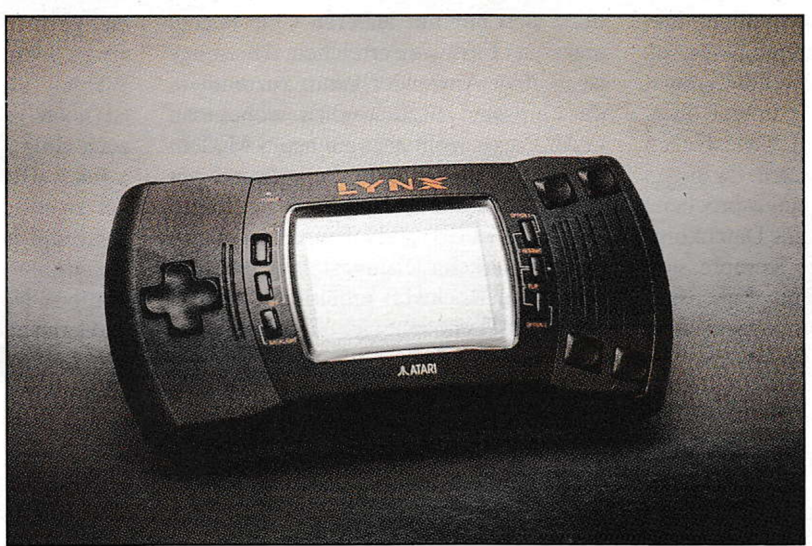
Modem: 748,- DM, 6 Monate Garantie
FAX-Software für ATARI: 60,- DM/138,- DM
kostenloser Update-Service

DTP Service Julian Riedlbauer
Tannenweg 12
W-4505 Meerbusch 1
Tel.: (02159) 81901

Modem: 799,- DM, 2 Jahre Garantie
inklusive FAX-Software für MS-DOS bzw. MS Windows
kostenloser Update-Service

LYNX II

Ataris tragbarer Entertainer



Anfang der Achtziger machte das VCS 2600 Atari zum Marktführer unter den Videokonsolenherstellern von Videospielen. Kurze Zeit später trat der Heimcomputer seinen Siegeszug durch die Kinderstuben an, und die Konsolen verschwanden in der Versenkung, bis die japanischen Konzerne Sega und Nintendo mit High-Tech-Hardware und Spitzenmodulen wieder Wind in die Szene brachten.

Dem Erfolg von Megadrive, Mario und Co. wollte Sam Tramiel von Atari nicht tatenlos zusehen. Also meldete sich Atari Ende 1988 mit dem Lynx im Konsolengeschäft zurück. Ohne durchschlagenden Erfolg allerdings: Technisch steckte der farbige Mobilentertainer seinen direkten Konkurrenten Game Boy locker in die Tasche. Jedoch floß der Modulstrom über lange Zeit so zäh wie Sirup dahin, und auch der enorme Batterieverbrauch machte die edle Kiste nicht gerade attraktiver. So mancher Experte sah wohl schon das Aus fürs Lynx, doch Atari wußte zu kontern. In Silicon Valley wurden dem Handheld alle Kinderkrankheiten ausgetrieben. Für rund DM 200,- hält der Käufer nun ein quasi runderneuertes Gerät in den Händen.

Facelifting

Gleich bei der ersten Begegnung mit dem neuen System fällt auf, wie angenehm der Controller in den Händen liegt. Neben einem kleineren Gehäuse haben die Entwickler dem Lynx II auch ein fetzigeres Outfit in dunkelblau und schwarz verpaßt. Am rund drei Zoll großen Farb-Display, dem großen Joypad und den vier Action-Buttons ging das Facelifting dagegen spurlos vorüber. Gut so, denn dadurch kommen Links- und Rechtshänder nach wie vor gleichberechtigt zum Zug. Durch Drükken von Pause und FLIP läßt sich nämlich jederzeit der Bildschirm umdrehen. Verrenkungen wie am Game Boy sind da nicht zu befürchten.

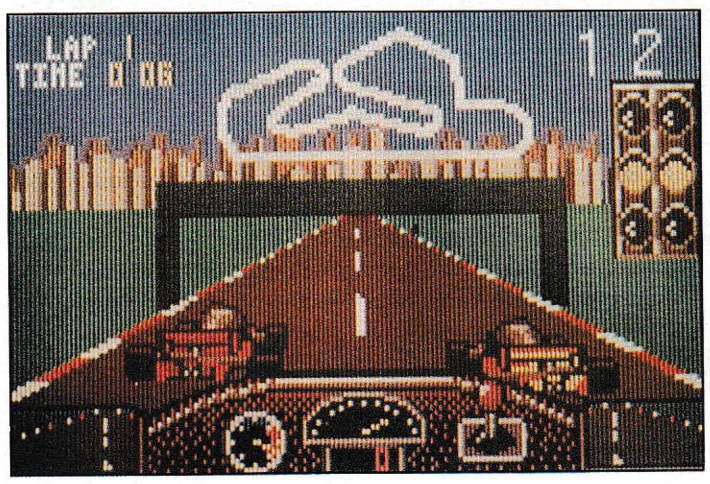
An der Rückseite des Geräts bietet der Batterieschacht Platz für sechs Mignonzellen. Herkömmliche Alkali-Batterien verrichten dort bis zu vier Stunden ihren Dienst, Akkus halten den Spielspaß rund 25% länger am Laufen. Um kostbare Energie zu sparen, kann die Bildschirmbeleuchtung ausgeschaltet werden. Zu empfehlen ist dies aber nur an hellen Tagen, ansonsten tränen einem schon nach ein paar Minuten angestrengten Guckens die Äuglein. Wer öfters in Steckdosennähe zockt, legt sich am besten für DM 25,- einen Netzadapter zu.

Lynx sucht Anschluß

Gleich neben der dafür vorgesehenen Anschlußbuchse liegt ein standardmäßiger Kopfhöreranschluß. Ist die Verbindung über Klinkenstecker erst einmal hergestellt, verwöhnt das Lynx die Lauscher mit Stereo-Sound aus vier voneinander völlig unabhängigen Kanälen. Aus dem internen Mini-Lautsprecher dringt dagegen nur monotones Gepiepse. Selbiges würgt man am besten umgehend am Lautstärkeregler ab, denn gerade in der U-Bahn sind Störenfriede nicht gern gesehen. Von der Dudel-Akustik einmal abgesehen, gibt sich das Lynx enorm kommunikationsfreudig. Über

Datalink können bis zu acht Geräte gekoppelt werden. Etliche Cartridges, wie beispielsweise das eher schlichte Formel-Eins-Rennen „Chequered Flag", laufen erst im Datalink-Betrieb zur Hochform auf.

Genau dieses Modul nutzt auch den speziellen 3D-Chip im Lynx aus. Mit seiner Hilfe ist es möglich, Objekte stufenlos bis auf Bildschirmgröße hochzuzoomen und beliebig zu drehen. Ansonsten versetzt das Lynx technisch keine Bäume. Im Innern der Konsole verrichtet ein 6502 im 4-MHz-Takt seine Dienste - jener Chip, der schon den Commodore 64 auf Touren hielt. Für seine Pixel-Kunstwerke in einer Auflösung von 160 * 102 Punkten steht dem Grafiker eine Palette von 4096 Farben parat. Gleichzeitig zu sehen sind davon aber maximal 16.

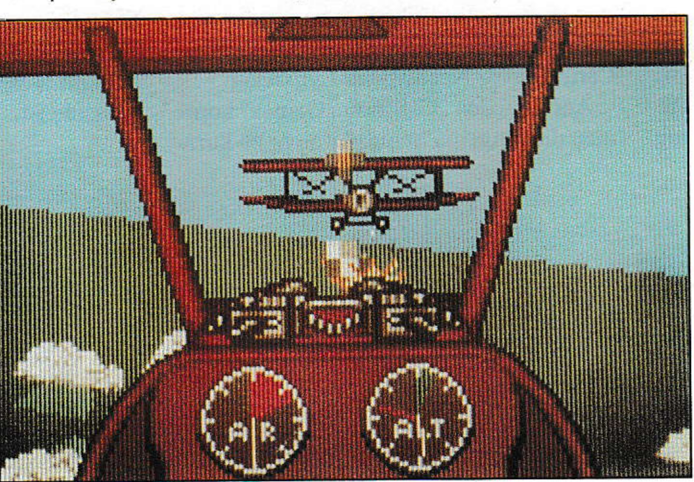


Checkered Flag

Faites votre jeux

Jetzt aber endlich konkret zu dem, was da eigentlich so über den Screen flimmert. Etwa 50 Module zu Preisen zwischen DM 50,- und 70,- buhlen derzeit um des Gamblers Gunst. Bis Ende des Jahres sollen noch einmal 30 dazukommen. Das Gros des Angebots stellen Umsetzungen von Spielautomaten. Kein Wunder, setzt Atari doch seit über einem Jahrzehnt immer wieder Akzente im Arcade-Bereich. Was aber noch lange nicht bedeutet, daß die Cartridge-Versionen genauso viel Spaß wie ihre Vorbilder auf den Schaltkreisen beheimaten. Ganz im Gegenteil. Module wie die monotone Zwei-Spieler-Ballerei „Xenophobe" oder die langatmige Ganovenhatz „A.P.B." gehören klar in die Kategorie Ausschußware. Eine etwas bessere Figur macht da schon „Rygar", auch wenn dieses angestaubte Hack'n' Slay-Epos außer flotten Sprites und einem gesalzenen Schwierigkeitsgrad dem Genre keinerlei neue Impulse verleiht. Ebenfalls recht bodenständig kommt „Paperboy" daher. Auf dem BMX-Radel hat der Spieler dafür zu sorgen, daß die Abonnenten jeden Morgen ihre Tageszeitung im Briefkasten haben. Am Anfang macht das Pedaletreten auf dem scrollenden Stadtparcours noch Spaß. Doch auch die eingestreuten Zwischen-Levels auf der Cross-Strekke zögern die ersten Gähnanfälle nur um wenige Minuten hinaus.

Auch aus der Idee von „Pac Land" hätte Atari mehr herausholen können als ein ödes Jump and Run. Videospiel-Veteran Pac Man läuft von links nach rechts durch eine extrem naiv gezeichnete Comic-Welt, weicht Geistern aus und sammelt Früchte ein. In höheren Levels scrollt das Lynx tadellos in mehreren Ebenen. Ein hübscher Effekt, der die Verweildauer im Modulschacht ein wenig erhöht. Wer von Pacman und Konsorten nicht genug kriegen kann, greift aber lieber zur originalgetreuen Adaption von „Ms. Pac Man". Mit rosa Schleife im Haar rast die gefräßige Heldin durch ein Labyrinth voller Pillen, immer auf der Flucht vor einer Horde Gespenster. Fast genauso viele Jährchen auf dem Buckel wie Fräulein Pac haben die vier Riesenmonster aus der Verwüstungsorgie „Rampage". Als Gorilla oder Rieseneidechse macht der Spieler nach und nach diverse amerikanische Millionenstädte dem Erdboden gleich. Natürlich sieht die Bevölkerung den rapiden Verfall der Bausubstanz nicht gerade gern und hetzt Panzer, Polizei und MG-Schützen auf die riesigen Zerstörer. Obwohl moralisch nicht ganz unbedenklich, macht Rampage auch heute noch mächtig Laune - gutes Spieldesign zahlt sich eben aus. Gleiches gilt für den Oldie „Gauntlet", auf den die Bezeichnung Mini-Rollenspiel wohl am ehesten paßt. Ein, zwei oder vier Recken ziehen los, um in über 20 scrollenden Irrgärten nach Schlüsseln und Schätzen zu suchen. Dabei begegnen ihnen allerlei gruftige Gesellen und sogar Gevatter Tod höchstpersönlich. Durch die Link-Option geht im Mehr-Spieler-Modus mächtig die Post ab. Zusätzlich heizt die abwechslungsreiche Grafik die heißen Konkurrenzkämpfe um den höchsten Score an.



Warbirds

Für Leute, die ihren Adrenalinspiegel lieber am Steuer eines Boliden hochpeitschen, bietet „Chequered Flag" eine solide Formel-1-Simulation mit zahlreichen Rennen auf flotten 3D-Pisten. Gerade wenn mehrere Nachwuchspiloten ihre Runden drehen, fällt der Mangel an Tuning-Optionen und Boxenstops nicht weiter auf. Dagegen steckt im Sportwägelchen aus „Roadblasters" echtes High-Tech-Equipment. Und das ist auch bitter nötig, denn in dieser militanten 3D-Hatz darf nicht nur nach Herzenslust aufs Gaspedal getreten, sondern auch auf den Feuerknopf eingehämmert werden. Stattliche 50 Levels erwarten den stahlharten Driver. Hindernisse auf der Fahrbahn, Scharfschützen am Streckenrand und ein enormer Spritverbrauch sorgen für ein kurzes Bildschirmleben. Zum Glück wirft ein Helikopter hier und da Extrawaffen ab - das Maschinengewehr im Anschlag, ist einem jedes Tempolimit schnurzpiepegal.

Ataris bislang beste Automatenumsetzung kommt allerdings ganz ohne Brutalität aus. Zurecht genießt „Klax" unter Lynxianern in etwa den gleichen guten Ruf wie Nintendo-Tetris für den Game Boy. Wie immer bei genialen Tüftelspielen ist die Idee denkbar einfach: Von oben herab rollen farbige Klötze auf einem Fließband heran. Unten angekommen, fängt sie der Spieler mit einer Art Gabelstapler auf und deponiert die Ladung in einen Behälter. Durch überlegtes Schichten entstehen Reihen aus drei oder mehr Blöcken gleicher Farbe. Da auch horizontale und diagonale Linien zählen, sind echte Domino-Effekte möglich. In den 100 ausgefeilten Levels steckt mehr Suchtgefahr als in hundert Schachteln Zigaretten. Zudem geizt das Modul nicht mit schicker Sprachausgabe und einer Batterie zum Speichern von High-Scores.

Ebenfalls in die Grübel-Kerbe schlägt „Block Out". Das Lynx eigene Tetris-Surrogat kleidet das be-

kannte Spielprinzip in ein fesches 3D-Ambiente. Von oben herab fallen geometrische Objekte in einen Becher hinein, dessen Standfläche und Höhe der Zocker genau wie den Schwierigkeitsgrad vor jeder Partie festlegt. Läßt man die Teile nun so rotieren und in den Behälter purzeln, daß ebene Flächen entstehen, werden diese wie von Geisterhand weggenommen. Räumliches Vorstellungsvermögen und gute Reflexe sind der Schlüssel zum Erfolg. Jedermanns Sache ist „Block Out" aufgrund der sehr abstrakten Grafik sicher nicht, Nachwuchs-Einsteins bringt es aber jede Menge Spaß. Nicht minder stark gefordert werden die grauen Zellen bei „Chip's Challenge". Um seiner neuen Herzdame zu imponieren, geht der kleinwüchsige Held Chip ein großes Abenteuer in 144 Levels ein. In jedem dieser scrollenden Labyrinthe liegen Speicherbausteine versteckt. Sie gilt es aufzusammeln, und dann nichts wie ab zum Ausgang. Leider machen Feuerwände, Eis- und Magnetfelder sowie unüberwindbare Wassermassen diese Aufgabe zum Drahtseilakt für flinke Denkprofis. Eine sehr detaillierte Grafik und Paßwörter zum direkten Einstieg in höhere Levels entschädigen den Byte-Experten für so manch verschwendete Gehirnzelle. Vom gleichen Entwicklerteam wie „Chips Challenge" stammt das eher actionlastige Irrgartenspielchen „Electrocop". Die Knarre im Anschlag, erkundet der Spieler in Robocop-ähnlicher Montur das Hauptquartier eines Industriegiganten. Neben bösen Buben stehen auf den Gängen auch ein paar ganz witzige Spielautomaten mit Klassikern wie Defender oder Breakout herum. Von dieser Idee und der beeindruckend flüssigen 3D-Grafik einmal abgesehen, bietet der Bulle mit Stromanschluß leider nur magere Hausmannskost.

Ebenfalls im Polygon-Look kommen die beiden Flugzeugspielchen „Blue Lightning" und „Warbirds" daher. Letzteres nimmt gar für sich in Anspruch, der erste waschechte Flugsimulator fürs Lynx zu sein. „Blue Lightning" hingegen ist eine starke Variante des Afterburner-Prinzips: drei Luft- und Bodenangriffe in Perfektion. Leider sind die Levels ein wenig lang geraten. Außerdem fliegt man häufig durch die Gegend, ohne daß etwas passiert. „Warbirds" schneidet bei näherer Betrachtung kaum besser ab. Einige Zeitlang sorgen die Doppeldecker-Dogfights für gepflegte Erste-Weltkriegs-Atmosphäre, vor allem, wenn mehrere Lynx-Piloten via Modem-Link gleichzeitig in die Luft gehen. Das Aus für das moralisch nicht ganz standfeste Vergnügen folgt aber spätestens in den übertrieben schwierigen höheren Abschnitten.

Auf ganz andere Art unappetitlich geht es in der „Slime World" zu. Mit Jetpac und Laserwaffe laufen bis zu acht furchtlose Höhlenforscher durch verwinkelte Irrgänge. Massig eklige Gegner, unter ihnen Riesenspinne und Freund Fledermaus, schleimen durch die Gewölbe. Ideales Kanonenfutter für das stattliche Arsenal von Extrawaffen. Wer's noch eine Stufe rauher liebt, greift am besten zur Ballerware für unterwegs. „Zarlor Mercenary" und „Gates of Zendocon" befriedigen die Ballergelüste schießwütiger Joypad-Artisten auf passable Art. Neben dem Verteidigungs-Equipment und der Level-Anzahl unterscheiden sich die Programme nur durch die Richtung, in die das Display scrollt. Während die „Gates of Zendocon" in bester R-Type-Manier zur Seite hin offen stehen, schlägt sich der Söldner aus Zarlor von oben nach unten durch Feindesland. Auch der bonbonbunte Geschicklichkeitstest „Super Skweek" aus Frankreich scrollt fleißig vertikal. Als liebliches Knuddelwesen färbt der Spieler auf sieben Inseln mehr als zwanzig Levels rosa ein - was kein größeres Problem wäre, wuselten nicht allerhand kauzige Gesellen auf den Plattformen herum. Gottlob gibt es ein paar sinnvolle Schußwaffen sowie Teleporterfelder, Abkürzungen und Geheimgänge in höhere Abschnitte, die einen vor so dem frühzeitigen Game Over bewahren. Auf der technischen Seite bietet Super Skweek keinen Anlaß zur Kritik: Comic Grafik und liebliche Melodien erfreuen den Betrachter. Dank des moderaten Schwierigkeitsgrades genau das richtige Cartridge für Nachwuchs-Lynxianer.

Auch bei den „California Games" kommen die jüngeren Semester nicht zu kurz. Ob Surfen, BMX-Biking, Skateboard-Stunts oder Jongleurarbeiten mit dem Footback - immer sind exaktes Timing, eine präzise Steuerung und viel Ausdauer gefragt. Drei Jahre hat der Epyx-Klassiker nun schon auf dem Buckel, ohne auch nur ein Byte von seiner Spritzigkeit verloren zu haben. Mindestens genauso zeitlos: „Shanghai". In japanischen Teehäusern seit dem vierten Jahrhundert als Mah-Jong für seinen hohen Suchtfaktor bekannt, sorgt auch die neuzeitliche Modulfassung für jede Menge Motivationsschübe. Auf dem Screen erscheinen 144 asiatische Ornamente. Gleiche Symbole gehören zusammen und können durch Anklicken entfernt werden. Clou dabei: Die Steinchen liegen in verschiedenen Figuren, z.B. Drachen, Spinne etc., in mehreren Ebenen übereinander. Abtragen in der richtigen Reihenfolge ist also angesagt. Auch auf der niedrigsten Schwierigkeitsstufe ohne Zeitbegrenzung eine Herausforderung, von der keiner so schnell die Finger läßt. Damit schließt sich der Reigen. Es gibt natürlich noch mehr Module fürs Lynx, denn Atari hat für jeden Geschmack und jedes Alter wenigstens ein passendes Game im Repertoire.

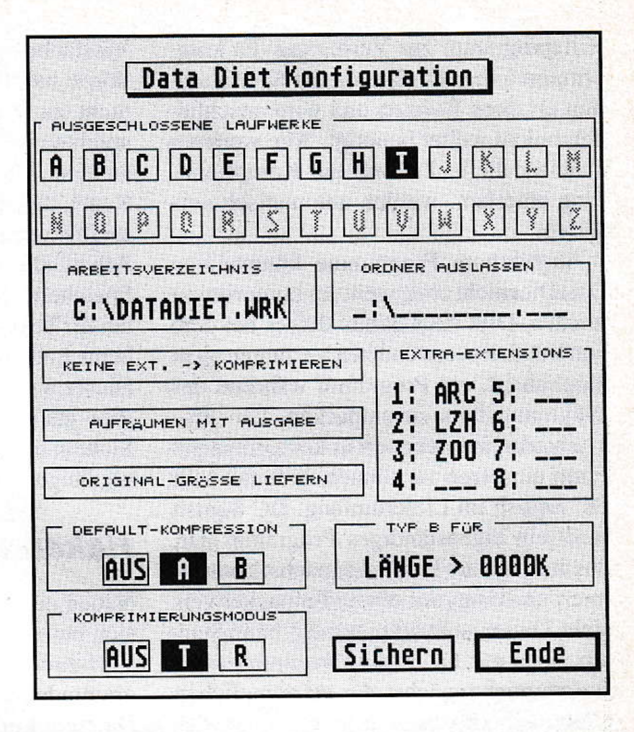
Vorausschau

Attraktivitätssteigernd feiern gleich mehrere Computerspielklassiker im Herbst ihr Debut auf dem Handheld: Ab Oktober turnen die todesmutigen Lemminge durch 100 knifflige Landschaften, vor Jahreswechsel steht mit „Eye of the Beholder" das erste Rollenspiel für den portablen Entertainer ins Haus. Damit sollte sich das Lynx endgültig vom Insidertip zum begehrten Videospielsystem mausern. Verdient hätte das die durchdachte Konsole allemal. Atari hat zweifellos die beste portable Videospielkonsole, wenn man sie mit Segas Game Gear oder Nintendos Game Boy vergleicht. Nun müßte Ataris Marketing-Abteilung nur noch aus ihrem tiefen Schlaf erwachen und durch aggressive Werbung und Medienarbeit der Öffentlichkeit erzählen, wie gut die hauseigene Konsole ist.

CBO

Dateien auf Diät

DataDiet



Programme zum Packen von Daten auf Online-Basis sind inzwischen in Mode zu gekommen. Bereits in [1] war mit DataLight ein Programm zur Online-Komprimierung von Dateien auf dem Prüfstand. Daß man die gleiche Zielsetzung auch auf andere Art und Weise verwirklichen kann, zeigt DataDiet. Dieses Programm verwendet ein grundsätzlich anderes Verfahren zum Packen der Dateien als DataLight.

Noch einmal kurz zur Erinnerung: DataLight arbeitet beim Packen von Daten block-orientiert, also auf BIOS-Ebene. Mehrere Sektoren eines Mediums werden dabei als eine Einheit angesehen, egal ob es sich dabei um Verwaltungssektoren (also FAT oder Directory) oder um Datensektoren handelt. Beim Zugriff auf die komprimierten Daten werden diese gelesen, dekomprimiert und in dieser Form im Speicher gehalten. Bei Schreibzugriffen werden die Daten komprimiert und zurückgeschrieben Die Komprimierung von DataLight bezieht sich dabei stets auf alle Daten einer Diskette oder Festplatten-Partition.

Ein interessantes Konzept

DataDiet verfolgt hier eine andere Strategie. Es wird nicht das Speichermedium als Ganzes komprimiert, sondern jede Datei für sich. FATs und Directory bleiben somit unverändert, komprimiert werden lediglich Datensektoren. Bei jedem Dateizugriff wird geprüft, ob eine komprimierte oder eine unkomprimierte Datei angesprochen wird. Liegen komprimierte Daten vor, wird die komplette Datei zunächst vollständig ausgepackt und in einem speziellen Arbeitsverzeichnis in entpackter Form bereitgehalten. Alle weiteren Zugriffe auf die komprimierte Datei werden von DataDiet auf die unkomprimierte Kopie im Arbeitsverzeichnis umgelenkt. Beim Schließen der Datei oder bei Beendigung eines Programms wird diese Kopie dann wieder komprimiert und an die richtige Position geschrieben. Diese trickreiche Vorgehensweise hat den Vorteil, daß nach dem einmaligen Entpacken der Datei solange keine Zeitverluste durch Komprimierungsoperationen mehr auftreten, bis die Datei endgültig zurückgeschrieben wird. Da bei diesem Verfahren keine direkten Zugriffe auf einzelne Sektoren eines Mediums erfolgen, läßt sich die von DataDiet realisierte Art der Datenkompression auch in Netzwerken verwenden.

Installation

Kommen wir zur Installation von DataDiet. Los geht's mit dem Kopieren des DataDiet-Treibers in den AUTO-Ordner des Boot-Laufwerks. Da die Reihenfolge der Programme im AUTO-Ordner nicht ganz unwichtig ist, gibt das Handbuch hierzu einige Hinweise. Der wichtigste Punkt der Installation ist die Konfiguration von DataDiet. Die Programmbeschreibung wartet zunächst mit Hinweisen zu einer einfachen Installation auf, die die besonderen Möglichkeiten des Programms außer acht läßt und sich auf das Wesentliche beschränkt. Dies dürfte besonders für denjenigen nützlich sein, der sich mit der Datenorganisation auf Festplatten weniger gut auskennt.

Beim Konfigurieren von DataDiet wird in erster Linie festgelegt, welche Dateien auf welchem Laufwerk komprimiert werden sollen. Da DataDiet dateiorientiert arbeitet, hat man völlig freie Wahl bei deren Auswahl. Beispielsweise lassen sich bestimmte Laufwerke, Ordner oder Dateitypen von der Komprimierung ausschließen. Letzteres ist vor allen Dingen dann interessant, wenn sich bereits komprimierte Dateien (Archive) auf einem Laufwerk befinden. Diese lassen sich naturgemäß nicht weiter komprimieren und werden deshalb je nach Einstellung von DataDiet ignoriert. Schließlich wird noch das Arbeitsverzeichnis festgelegt, in dem sich später die Arbeitskopien der komprimierten Dateien befinden. Dieses Verzeichnis kann auch auf einer RAM-Disk eingerichtet werden, was Geschwindigkeitsvorteile gegenüber einer Festplatte oder gar Diskette bringen kann. Hat man sich für eine Grundeinstellung entschieden, kann man diese in Form einer editierbaren Textdatei sichern.

Für die Komprimierung stehen zwei Algorithmen zur Verfügung, die sich durch Geschwindigkeit und Effizienz voneinander unterscheiden. Sie ahnen es sicher schon: Der effizientere Algorithmus ist natürlich auch der langsamere. DataDiet erlaubt es, beide Algorithmen in Abhängigkeit von der Dateilänge nebeneinander einzusetzen. So lassen sich lange Dateien mit dem einen, kürzere Dateien mit dem anderen Verfahren komprimieren. Diese Einstellung läßt sich zu jedem Zeitpunkt wieder ändern, ohne neu booten zu müssen.

Hat man DataDiet den eigenen Bedürfnissen entsprechend konfiguriert (oder einfach die Standardeinstellung übernommen), ist das Programm nach dem nächsten Reset zwar aktiv ...

Komprimierung zum ersten

... aber natürlich ist zu diesem Zeitpunkt noch gar nichts komprimiert. Um die in der DataDiet-Konfiguration festgelegten Dateien erstmalig zu packen, steht ein

Hilfsprogramm zur Verfügung. Es komprimiert lediglich alle von der Konfiguration erfaßten Dateien und wird anschließend nicht weiter benötigt. Alle weiteren Dateien, die in Zukunft zur Komprimierung anstehen, werden automatisch verarztet.

Ausführbare Programme können von DataDiet nicht ohne weiteres komprimiert werden. Dies liegt daran, daß es bei dem von DataDiet verwendeten Verfahren nicht möglich ist, ein Programm während des Programmstarts zu entpacken. Um Programmdateien dennoch in komprimierter Form einsetzen zu können, befindet sich DC Squish im Lieferumfang. DC Squish stellt ein eigenständiges Programm dar, das ausführbare Dateien zunächst komprimiert und dann mit einem Entpacker versieht. Dieser ist dafür zuständig, beim Starten eines mit DC Squish komprimierten Programms zunächst die ursprünglichen Daten zu entpacken und erst dann das eigentliche Programm zu aktivieren. Dieses Verfahren läßt sich allerdings ohne weiteres nur bei Programmen anwenden, die nicht im Rahmen ihrer eigenen Konfiguration ihren Programmcode verändern. Dies ist bei den weitaus meisten Programmen jedoch nicht der Fall.

Die Gretchenfrage

Hier geht es natürlich um die Geschwindigkeit, oder eher um den Verlust dergleichen beim Einsatz von DataDiet. Es ist nicht so einfach, hiervon einen objektiven Eindruck zu vermitteln. Ein einfacher Test wie in [1], bei dem lediglich eine große Zahl an Dateien kopiert wird, würde jedenfalls zu irreführenden Ergebnissen führen. Beim Kopieren brauchen schließlich keine Daten entpackt zu werden, was dazu führt, daß das Kopieren komprimierter Dateien aufgrund der kleineren Dateilängen schneller vor sich geht als im normalen Betrieb. Um also eine Aussage über die Geschwindigkeit der DataDiet-Routinen zu erhalten, muß man sicherstellen, daß die Daten zwischenzeitlich ge- oder entpackt werden. Hierzu werden Daten zwischen zwei Festplatten-Partitionen kopiert, wobei eine Partition nur unkomprimierte, die andere nur komprimierte Dateien enthält. Dies läßt sich durch eine geeignete Konfiguration von DataDiet realisieren.

Testläufe auf Atari ST und TT mit Festplatten unterschiedlicher Übertragungsrate zeigen, daß für das Kopieren größerer Datenmengen beim TT höchstens dreimal mehr Zeit benötigt wird als im Normalfall. Beim ST sieht es trotz der geringeren Taktfrequenz nicht viel ungünstiger aus: Hier steigt die Wartezeit im Schnitt auf ein Vierfaches an. Dabei hängen die genauen Werte natürlich stets davon ab, wie viele nicht zu komprimierende Dateien (beispielsweise Programmdateien) kopiert werden. Nun ist es allerdings nicht die Regel, daß man Megabytes an Daten hin- und herkopiert. Häufig beschränkt sich die Arbeit am Rechner auf das Laden und Speichern einzelner Dateien, so z.B. auch bei der Textverarbeitung. Wie sieht es also beim Laden einer großen Textdatei aus? Dieser Vorgang nimmt mit DataDiet die vier- bis fünffache Zeit in Anspruch, das Sichern der gleichen Datei dauert ein wenig länger.

Haken und Ösen

Neben einer Reihe von Vorteilen zeigen sich bei dem von DataDiet verwendeten Verfahren auch zwei Nachteile. Daß Programmdateien getrennt von den restlichen Dateien komprimiert werden müssen, wurde bereits angesprochen. Hinzu kommt ein Problem, das die Dateilängen betrifft. Komprimierte Dateien sind schließlich nur scheinbar kürzer. Im Klartext: Der Directory-Eintrag einer komprimierten Datei bezieht sich stets auf die physikalische Größe einer Datei, also auf den Platz, den diese tatsächlich benötigt. Einige Programme erfragen nun die Dateilänge über Funktionen des Betriebssystems, die sich auf die im Directory vermerkte Länge stützen. Da dieser Wert aber in der Regel zu klein ist, besteht die Gefahr, daß beim anschließenden Laden der Datei nicht genügend Zeichen berücksichtigt werden.

DataDiet bietet für solche Fälle die Option, den vom Betriebssystem gelieferten Wert für die Dateilänge zu beeinflussen, so daß in jedem Fall die „wahre" Länge einer Datei zurückgeliefert wird. Dies ist allerdings mit einem Zeitverlust bei Datenzugriff verbunden. Zwar wird in der Programmbeschreibung darauf hingewiesen, daß eine solche Korrektur der Dateilänge in der Regel nicht notwendig ist, aber der Verzicht auf die Dateilängenkorrektur birgt die Möglichkeit eines Datenverlustes. Der unbedarfte Anwender sollte hier Vorsicht walten lassen. Auch wenn DataDiet mit einer README-Datei ausgeliefert wird, die Programme aufzählt, die die korrekte Dateilänge benötigen, sollte man hier kein Risiko eingehen.

Von diesem Spezialfall abgesehen, ist die Datensicherheit bei Verwendung von DataDiet als hoch einzustufen. Dies folgt aus dem Umstand, daß jede Datei, während sie bearbeitet wird, weiterhin in komprimierter Form vorliegt und somit jeweils nur die im Arbeitsverzeichnis abgelegte Kopie bearbeitet wird. Die Gefahr eines Datenverlustes ist somit nicht höher einzustufen, als beim Arbeiten mit unkomprimierten Daten.

Schlußpunkt

Das von DataDiet verwendete Verfahren zur Online-Komprimierung verbindet eine akzeptable Geschwindigkeit mit hoher Datensicherheit. Besonders praktisch ist es, daß sich alle Einstellungen jederzeit durch ein Accessory verändern lassen, wobei sich die Änderungen sofort auswirken. Dadurch, daß man Dateien von der Komprimierung ausschließen kann, ist ein schneller Zugriff auf häufig benötigte Daten gewährleistet, so daß sich der für online-Komprimierer obligatorische Geschwindigkeitsverlusts in Grenzen hält. Was die Besonderheiten in Verbindung mit der Dateilänge und der Komprimierung von Programmdateien betrifft, lassen sich Schwierigkeiten bei der Handhabung von DataDiet jedoch nicht ganz ausschließen. Das informative Handbuch ist in einem gut lesbaren Stil geschrieben, der auch den weniger versierten Benutzer nicht überfordert. Das Prinzip, nach dem DataDiet arbeitet, wird recht ausführlich erklärt, so daß man nach dem Studium des Handbuchs einen guten Eindruck von der Funktionsweise des Programms gewonnen hat.

Als störend haben sich während der Testphase kleinere Mängel in puncto sauberer Programmierung erwiesen. So trägt DataDiet seinen cookie falsch ein, falls der cookie jar bereits voll ist. Außerdem scheinen die Redraw-Routinen der Dialogboxen in manchen Fällen nicht korrekt zu funktionieren. Wichtig für Besitzer des KAOS-TOS ist noch die Tatsache, daß DataDiet nicht mit dieser inoffiziellen, modifizierten Systemversion zusammenarbeitet. Auch unter MultiTOS läßt sich das Programm derzeit noch nicht einsetzen. Was den Preis betrifft: DataDiet ist für 129,- zu haben.

US

Bezugsadresse:

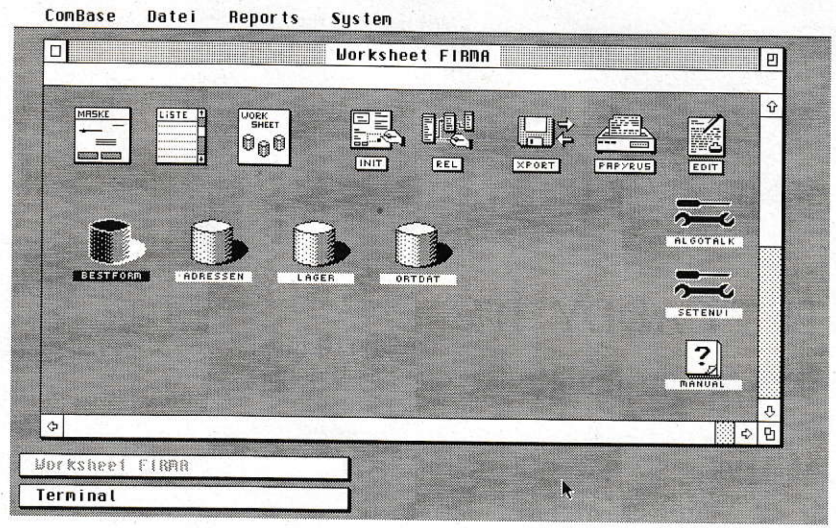
Artifex Computer GmbH
Anton-Burger-Weg 147
W-6000 Frankfurt 70
Telefon: (069)64688518

Literatur:

[1] „DataLight - Per Software aufgemotzt", ST-Computer 4/92

ComBase

Flexible Datenbank



Auffällig: Tempus Word mit neuer Benutzerführung

Vom Heim-Verlag wird ein Datenbanksystem unter dem Namen ComBase vertrieben. Die Entwicklung erfolgte unter Regie der COMPO Software GmbH. Es ist nicht das erste Datenbanksystem auf dem ATARI-Software-Markt, die Konkurrenz ist mittlerweile mächtig. Mit diesem Test soll geklärt werden, ob ComBase dieser Konkurrenz standhalten kann.

ComBase wird mit einer Diskette und einem schätzungsweise 500 Seiten starken Handbuch ausgeliefert. Das Handbuch führt in die Installation, die Grundlagen, die allgemeine Bedienung sowie die Verarbeitung von Algorithmen ein.

Bevor wir ComBase näher betrachten, schauen wir in das Tagebuch eines Anwenders, Entwicklers und Testers. Dort könnten folgende Eintragungen zu finden sein, die auch die ersten Schritte des Tests widerspiegeln.

„Erster Tag: Heute ist das Päckchen mit ComBase angekommen! Voller Neugier das Handbuch ausgepackt und flüchtig betrachtet. Ganz schön dick! Aber zu Anfang zuviel zu lesen. ATARI-Programme sind mittlerweile recht einfach zu bedienen, zumindest die grundlegenden Dinge klappen auch ohne Handbuch. Folglich die Diskette eingelegt, Verzeichnis geöffnet - und dann kam ein kleiner Schreck: drei Dateien auf der Diskette: LIESMICH.TXT, COMBASE.TOS und FLASHACC.TOS. Naja, LIESMICH.-TXT kann man ja mal lesen. Es könnte ja etwas Wichtiges darin stehen. Erst einmal kommt eine lange Liste von Bugfixes und Neuerungen diverser Versionen, obwohl mich eigentlich gar nicht interessiert, was bei irgendwelchen alten Versionen so alles falsch war. Unter anderem finde ich glücklicherweise den Hinweis, daß die Installation so erfolgen kann, wie es im Handbuch beschrieben wird. Da ein Installationsprogramm fehlt, ziehe ich das Handbuch zu Rate. Auf Verdacht kopiere ich die beiden Programmdateien schon mal auf die Festplatte. Die Vermutung liegt nahe, daß diese gepackt sein könnten.

Dort steht auch, daß die bereits kopierten Dateien genau richtig waren. Zwar heißt LIESMICH.TXT im Handbuch LIESMICH, aber das macht ja nichts. Soweit, so gut. Wie ich aus den Installationshinweisen erfahre, besteht ComBase aus zwei Teilen: ComBase und FlashAccess. Ich installiere erst einmal den ersten Teil, denn es steht ja nicht da, daß das Accessory *FlashAccess* (der Datenbankkern) auch installiert werden soll.

Nach einigem Kopieren ist ein Verzeichnis COMBASE auf der Festplatte angelegt. Voller Neugier starte ich das Programm. Kurz nach dem Laden meldet sich das Programm mit „FlashAccess ist nicht installiert! ComBase trotzdem starten? (Führt zu BOMBEN-ABSTURZ bei Datenbankzugriff!)“. Ich denke mir nur, daß sich hier noch einige Anwender freuen werden, wenn das Programm freundlicherweise mit den Atari-üblichen Bömbchen abstürzt. Ich versuche es trotzdem einmal, denn ich kann nicht glauben, daß die Fehler nicht abgefangen werden. Ein Menü erscheint, ich klicke einen Eintrag an - und tatsächlich, das Programm stürzt ab! Also neu booten und FlashAccess neu installieren ...

Nach dem Entpacken der Datei FLASHACC befindet sich ein Ordner namens FA_SYS auf der Festplatte. Darin ist ein Programm, welches auf eine Konfigurationsmöglichkeit schließen läßt. Also starte ich dieses. Es erscheint ein ungewöhnliches, meiner Meinung nach nicht gerade schönes Menü. Mit dem Programm kann ich aber so auf den ersten Blick nichts anfangen, also verlasse ich es wieder.

Nun aber der wichtige Schritt: FA22.-ACC wird auf Laufwerk C: kopiert und wieder neu gebootet. Aber beim Hochfahren des Rechners erscheint die Meldung, daß eine Resource-Datei fehlt. „Wieso fehlt einem Accessory eine Resource-Datei?“ frage ich mich, denn seit TOS 1.04 dürfte den Entwicklerinnen und Entwicklern klar sein, daß ein Accessory keinen dynamischen Speicher langfristig reservieren darf, also auch keine Resource-Dateien laden darf. Beim Verlassen der alternativen Shell GEMINI erscheint seit der Nutzung von FlashAccess auch die Meldung, daß noch zwei Dateien geöffnet seien. (Wie sich bei einem Telephonat mit COMPO später herausstellt, hält FlashAccess ständig zwei Dateien geöffnet.)

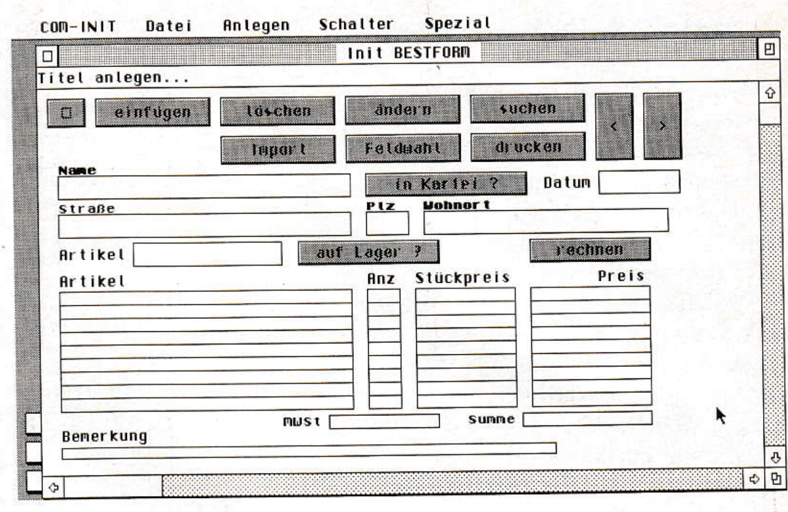
Nun suche ich nach der Resource-Datei, kopiere sie nach C: und hoffe, daß alles gut geht. Also neu gebootet und der nächste Versuch. Beim Start des Rechners stellt das Accessory nun fest, daß irgendeine Systemdatei fehlt. Auf Verdacht kopiere ich diese auch noch nach C:. Ich erhalte

nun zwar schon wieder eine Fehlermeldung, die (in englischer Sprache) fragt, ob ich denn eine neue Systemdatei anlegen wolle, aber egal, ob ich diese wünsche oder nicht, es erscheint anschließend die gleiche Fehlermeldung. Mein Drang, diesen Fehler zu ergründen, hält sich in Grenzen.

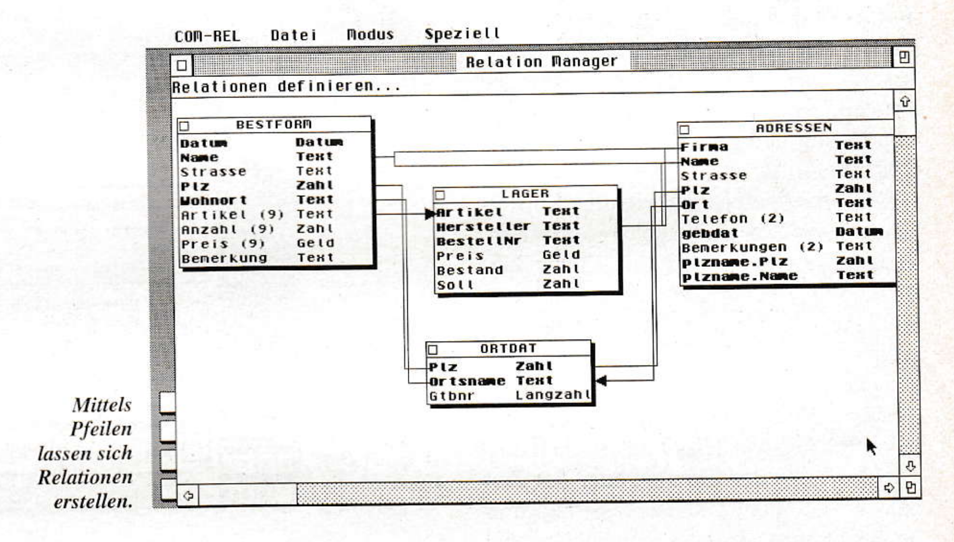
Nach nunmehr einigen Anläufen gelingt es mir tatsächlich, ComBase zu starten. Es erscheint ein Fenster, ein Menü und auf dem Desktop ein Text mit dem Titel „Terminal". Ich schaue erst einmal überall herum, die Texte auf dem Desktop werden immer mehr. Ich schließe die Fenster. Die Texte auf dem Desktop bleiben und im Menü nur der Titel SSWiS, aber kein Menüpunkt zum Verlassen des Programms. Auf Verdacht klicke ich auf die Texte und siehe da, die Fenster kommen wieder! Bei dem Init-Modul gibt es zwar ein Menü, in dieser Einstellung jedoch kann man die Menütitel zwar anklicken, aber leider erhält man nur einen Hinweis, daß die Drop-down-Menüs nicht aktiv seien. Warum ich die Einträge in den Drop-down-Menüs nicht wählen darf, bleibt mir verborgen. Nach dem Öffnen fast aller Fenster und der Anwahl eines Menüpunktes „Verlassen" bleiben noch zwei Texte auf dem Desktop. Der eine ermöglicht es mir nicht, ein Fenster durch Doppelklick auf den Text zu öffnen und der andere heißt „Terminal". Unter Terminal finde ich noch ein Fenster mit einer Begrüßungsmeldung und einigen Fehlermeldungen, die dort angezeigt werden. Da mir der Text nicht klar ist, schließe ich das Fenster wieder. Mit Control-Alternate-Delete verlasse ich das Programm und boote wieder neu.

ComBase ist auch für den TT gedacht, also betrachte ich den Programm-Header, um einmal zu sehen, ob die Bits für den alternativen Speicher meines TTs gesetzt sind. Immerhin möchte ich die 6 MB (2 MB ST-, 4 MB alternativen Speicher) auch ausnutzen. Sie sind es nicht, also setze ich sie. Danach starte ich wieder ComBase, um zu sehen, ob das Programm schneller läuft. (Der alternative Speicher auf dem TT ist schneller als der ST-Speicher.) Aber nun weiß ich, warum die Bits nicht gesetzt waren: das Programm stürzt direkt nach dem Laden ab. Nach einigem Probieren gelingt mir die Feststellung, daß ich zwar Speicher aus dem alternativen RAM anfordern darf, das Programm selbst muß jedoch im ST-Speicher laufen. Naja. Für den ersten Tag ist mir die weitere Lust auf einen neuen Test vergangen, ich werde mich doch erst dem Handbuch widmen.

Zweiter Tag: Die grobe Lektüre des Handbuchs ergab, daß das Accessory FlashAccess direkt auf C: installiert werden muß. Das behagt mir zwar nicht, da ich nie auf C: (als Boot-Partition) direkt installiere, aber FlashAccess funktioniert dann wenigstens.



Eine Maske wird erstellt.



Mittels Pfeilen lassen sich Relationen erstellen.

Als ersten Schritt wähle ich aus dem Kapitel „Konzeption" das Anlegen der ersten Datei, um ComBase kennenzulernen. Alles klappt gut, etwas arg störend ist jedoch gleich beim Anlegen des Beispielkommentars, daß auf meinem TT in mittlerer Auflösung partout der Cursor nicht in den Masken erscheinen will. Außerdem kann ich den Zeichensatz nicht einstellen, obwohl Zeichensätze via GDOS geladen sind. Und in der Dateiauswahlbox schlägt mir ComBase immer einen leeren Dateinamen (kein Name, nur eine Extension) vor, obwohl Namen dieser Art gar nicht legal sind. Irgendwie macht das INIT-Modul einen nicht ausgereiften Eindruck.

Mit etwas Fleiß ist das erste Ziel erreicht! Die erste Datenbank ist fertig. Es war zwar manchmal etwas Sucherei, was wo einzustellen ist, aber geklappt hat es! Auch die Eingabe der Werte klappt. Nun kann es endlich losgehen ...".

Soweit zum Tagebuch. Betrachten wir ComBase nun näher. ComBase basiert auf dem Fensterverwaltungssystem SSWiS. Das Fensterverwaltungssystem ist auch vom SPC Modula-2 und dessen Nachfolger her bekannt. Es erfordert etwas Umgewöhnung, denn die Vorteile eines Quasi-Multitaskings über SSWiS (basierend auf den Modula-2-eigenen Coroutinen) werden von Nachteilen überschattet. Die Bedienung des Programms ist nicht so, wie man es von GEM-Programmen, die sich an die etablierten Standards halten, gewohnt ist. Der erste Schritt einer Anwenderin oder eines Anwenders ist also die Gewöhnung an das modular aufgebaute Datenbanksystem.

ComBase ist unterteilt in die Module INIT zum Erstellen einer Datenbank, REL zum Aufbau von Relationen, MASK zur Bearbeitung der Daten, LIST zur Erstellung von Listen, XPORT zum Transferieren von Daten und PAPYRUS zur formatierten Ausgabe von Daten. Außerdem sind ein Manual und ein Editor dabei. Diese Vielzahl von Modulen wird über die SSWiS/ComBase-Oberfläche angesteuert.

Das INIT-Modul...

... ist das Modul, mit dem man zuerst Kontakt hat. Hier wird interaktiv die Datenbank angelegt. Deren Anlegen erfolgt über eine Maske, die eine 1:1-Darstellung

der Datenbank ist. Die Masken sind deren zentraler Bestandteil. Neben diesen 1:1-Masken ist es jedoch auch möglich, sogenannte Clone-Masken anzulegen, die nur noch einen Teil der Daten oder zusätzlich Daten einer anderen Datenbank beinhalten. Außerdem lassen sich virtuelle Masken generieren, die unabhängig von einer Datenbank zur Verfügung stehen.

Als Datentypen stehen Text, Zahl, Langzahl, Datum, Zeit, Dezimal, Geld, Boolean und Extern zur Verfügung, wobei der Geldtyp nur eine Variante des Dezimaltyps ist. Der Datentyp Extern dient zum Starten von SPC-Modula-2-Programmen, sogenannten Jobs. Ein Datentyp BLOB (Binary Large Object), der mehr und mehr Verbreitung findet, ist ebensowenig zu finden wie ein Datentyp für Bilder, die jedoch über Jobs eingebunden werden können. Die übliche Zusammenfassung von Datum und Zeit zu einem Zeitpunkt fehlt ebenso.

Für Texte lassen sich wichtige Einstellungen wie Farbe und Zeichensatz treffen, Formatwünsche insbesondere für die Zahlentypen bleiben jedoch offen. Die Auswahl eines Zeichensatzes war im Test jedoch trotz über GDOS geladene Zeichensätze nicht möglich.

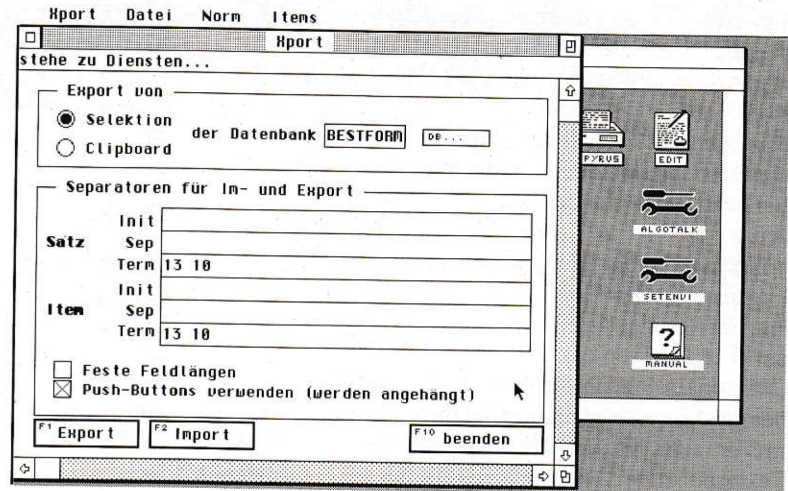
Makro-Buttons ermöglichen, bestimmte Funktionen wie den automatischen Eintrag des aktuellen Datums oder der Zeit fest zu verdrahten. Dies ist praktisch, gerade wenn man bedenkt, wie häufig bei bestimmten Anwendungen nur das aktuelle Datum einzutragen ist.

Zur Auflockerung der Masken wartet ComBase mit einem Grafikteil auf, der das Einfügen von Grafikflächen (Rechtecke mit oder ohne abgerundeten Ecken, mit diversen Füllmustern und Farben) ermöglicht. Bilder oder andere Grafikfiguren lassen sich nicht in die Masken aufnehmen.

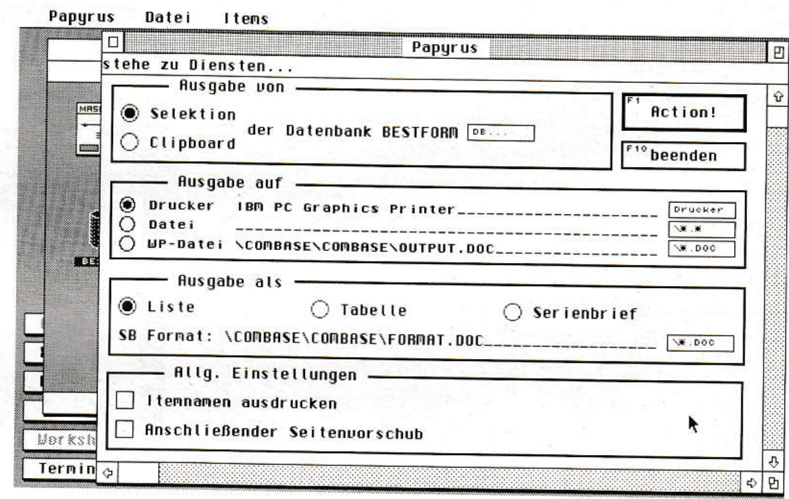
Der Funktionsvorrat beschränkt sich nicht nur auf die Felder und Grafikflächen, sondern reicht sogar bis zu Push-, Radiobuttons und Checkboxen, die für bestimmte Funktionen und Aufgaben praktischer sind als eine Texteingabe. Der Aufbau der Datenbank verfügt über alle notwendigen Möglichkeiten (inklusive der Indizierung).

Relationen

Mit dem Modul REL werden verschiedene Datenbanken miteinander verknüpft. Die Verbindungen - in ComBase auch Relationen genannt - erfolgen zum Zwecke des Datenaustauschs und des Mischens. Verbinden lassen sich alle Datenbanken, die in sogenannten Worksheets zusammengefaßt sind.



Vielfältige Möglichkeiten beim Im- und Export von Daten



Das Ausgabemodul Papyrus

ComBase verfügt über zwei Arten von Relationen: zum Verzweigen von einem Schlüsselfeld einer Datenbank in das einer anderen und zum Exportieren aus der einen Datenbank in eine andere.

Verbindungen zum Erhalt der referentiellen Integrität fehlen. Sie sind also nicht in der Lage, beispielsweise Löschregeln wie bei relationalen Datenbanken aufzubauen. Die Verbindungen sind somit weniger leistungsfähig, als man sich dies von einer relationalen Datenbank erhoffen würde.

Der Aufbau der Verbindungen erfolgt mit der Maus innerhalb eines Fensters. Sie werden grafisch dargestellt. Mit dieser Methode ist es einfach, den Überblick zu bewahren.

Datenverwaltung

Das MASK-Modul ist nicht dazu da, andere Masken anzulegen. Diese Aufgabe hat bereits das INIT-Modul übernommen. Das MASK-Modul als zweiter zentraler Bestandteil von ComBase neben dem INIT-Modul ermöglicht die Verwaltung der Daten. Hierzu gehören alle wichtigen Aktionen wie Darstellung, Eingabe, Änderung und Suche.

Die Masken werden in Fenstern dargestellt, sogar ein Clipboard wird unterstützt. Es handelt sich dabei jedoch nicht um das GEM-übliche. Ein Datenaustausch mit anderen Programmen ist folglich nicht möglich.

Die Auswahl der zu bearbeitenden Datensätze erfolgt über ein Drop-Down-Menü, bei dem nur relativ wenig Wahlmöglichkeiten bestehen. Es kann über alle Schlüsselfelder auch in Verbindung mit Wildcards (* und ?) ausgewählt werden. Die Möglichkeiten werden noch durch sogenannte Tief- und Hochwassermarken erweitert, wobei diese die Angabe einer Unter- und einer Obergrenze ermöglichen. Die Kombination unterschiedlichster, unter anderen komplexer Auswahlkriterien, auch über Nicht-Schlüsselfelder ist nicht möglich.

Die Masken lassen sich voll über Tastatur steuern, was besonders geübteren Personen entgegenkommt.

Aufgelistet

Das LIST-Modul befähigt ComBase dazu, die Inhalte der Datenbanken als Liste auszugeben. Die Darstellung erfolgt in tabellarischer Form. Wie bei dem MASK-Mo-

LogiLex-Produkte

1ST-Lock 2.0

1st-Lock ist ein speicher- und resetresidentes Programm, das alle Daten beim Speichern verschlüsselt und beim Laden wieder entschlüsselt. Der Benutzer, der sich über ein Passwort authorisiert hat, kann das gesamte System wie gewohnt verwenden. Illegale Zugriffsversuche werden mit Datum und Uhrzeit abgespeichert und können wieder aufgerufen werden.
1st-Lock ist ein selbsttätig startendes resetresidentes AUTO-Ordner Programm. Die Verschlüsselung erfolgt dermaßen rasant, daß auch bei schnellen Festplatten keine spürbare Verzögerung auftritt. Zudem verhindert das Programm aktiv das Festsetzen von Bootsektorviren.
Das Programm ist seit zwei Jahren auf dem Markt und zeichnet sich durch seine Betriebssicherheit und seine rasante Geschwindigkeit aus.
1st-Lock unterstützt alle ST/STE/TT-Rechner, alle TOS-Versionen, alle Grafikmodi, alle Disketten (auch HD) und alle Festplatten, die mit Atari-kompatiblen Gerätetreibern betrieben werden.

189,- DM

1ST-Card 2.0

1st-Card ist eine Volltext-Datenbank, es werden also komplette Texte erfaßt. Anders als in üblichen relationalen Datenbanken kann jedes Wort unabhängig von seiner Stellung im Datensatz gefunden werden.
Auch die Suchgeschwindigkeit im Volltext kennt keine Grenzen In einer Datenbank, die die gesamte Bibel enthält, findet man alle Stellen, die ein gesuchtes Wort enthalten in weniger als einer Sekunde.
Einzelne Datensätze können individuell miteinander verbunden werden.
Nicht nur die Bedienung, nein auch die Programmierung von 1st-Card erfolgt ausschließlich über die Maus. Sämtliche Funktionen zur Nutzung und Erstellung eines Expertensystems lassen sich per Mausklick realisieren.
1st-Card wurde vom Bundesbildungsminister mit dem deutschen Hochschulsoftwarepreis 1991 ausgezeichnet.

298,- DM

Preise sind unverbindlich empfohlene Verkaufspreise

DATA*light* 2.0

DATAlight ist ein speicherresidentes Programm, das die Daten die gespeichert werden sollen komprimiert und sie beim Laden wieder dekomprimiert. Dem Benutzer bleiben diese Manipulationen völlig verborgen. Er bemerkt nur, daß auf eine 720 KB Diskette plötzlich 1500 KB oder auf eine 44 MB-Wechselplatte auf einmal 90 MB passen.
Ein entscheidender Faktor für eine solche Komprimierung ist natürlich die Geschwindigkeit. Da der Benutzer selbst eingeben kann, wieviel Rechenzeit für das Komprimieren verwendet wird ist auch das kein Problem.
Datalight unterstützt alle ST/STE/TT-Rechner, alle TOS-Versionen, jeden Speicherausbau, alle Grafikmodi, alle Disketten (auch HD) sowie alle Festplatten mit Atari-kompatiblen Gerätetreibern.
Updates für alle LogiLex-Programme auf die neueste Version gibt es kostengünstig bei der Fa. LogiLex unter Einsendung der Originaldiskette und eines Verrechnungsschecks.

129,- DM

Hiermit bestelle ich beim Heim-Verlag, Heidelberger-Landstr.194, 6100 Darmstadt

Ich zahle (zuzüglich 6,- DM Versandkosten Ausland 10,- DM) unabhängig von der bestellten Stückzahl

- ❑ per Nachnahme
- ❑ per Scheck

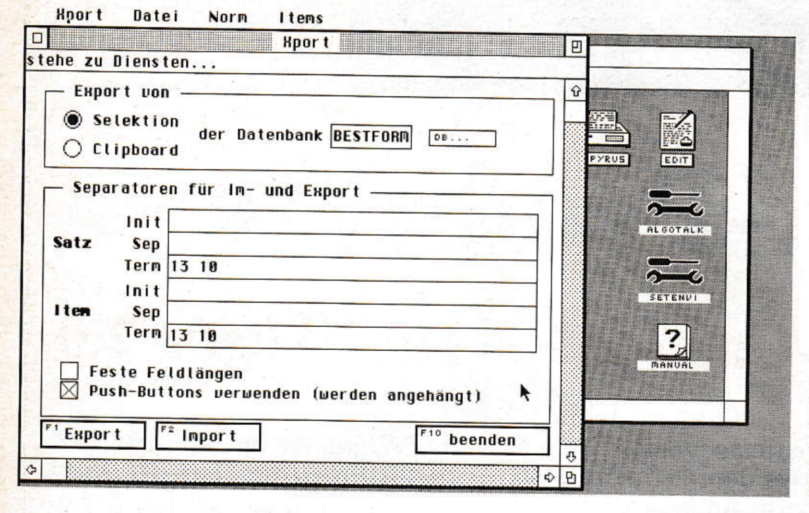
Name : ____________

Straße: ____________

Ort : ____________

Heim Verlag GmbH

Heidelberger-Land-Str. 194
6100 Darmstadt-Eberstadt
Tel.: 0 61 51 / 5 60 57 Fax: 5 60 59



Eine Datenbank-sprache ist in ComBase integriert.

dul ist auch hier die Angabe von Auswahlkriterien möglich. Praktisch ist die Einstellmöglichkeit der Schriftgröße, die jedoch nicht die Fähigkeiten des GEM widerspiegelt.

Der Datenim- und -export erfolgt über das XPort-Modul. Voraussetzung für die Nutzung dieses Moduls innerhalb eines Worksheets ist das Vorhandensein von Verbunden. Über XPort können jedoch auch Daten mit anderen Datenbanken ausgetauscht werden. Dem Datenaustausch steht also nichts entgegen, solange auch ComBase in der Lage ist, entsprechende Daten zu verarbeiten.

Die Einstellung der Parameter erfolgt individuell, eine Anpassung an die Datenformate diverser anderer Systeme geht problemlos vonstatten.

Ausgabe

Papyrus ist das Ausgabemodul von ComBase (und in diesem Falle nicht die Textverarbeitung eines anderen Software-Vertriebs). Es verarbeitet fremderstellte ASCII- und 1st Word Plus-Dateien. Als Serienbriefdruck steht der von 1st Word Plus zur Verfügung. Neben der Serienbrieffunktion kann die Ausgabe auch als Liste oder Tabelle erfolgen. Die Format-Dateien von 1st Mail (Serienbriefprogramm von 1st Word Plus) sind verwendbar. Neben den bekannten Formatanweisungen dürfen noch weitere in Papyrus zusätzlich integrierte wie die bedingte Ausgabe genutzt werden. Bei einfachen Serienbriefen bleibt kein Wunsch offen, mehr - beispielsweise Proportionalschrift - darf man jedoch nicht erwarten.

ComBase verfügt über einen eigenen Editor, der für diverse Zwecke benötigt wird. In erster Linie ist er zum Schreiben und Bearbeiten von Algorithmen gedacht, er eignet sich aber auch zum Beispiel zum Bearbeiten von Programmquelltexten.

Der Editor verfügt über alle notwendigen Eigenschaften eines Editors wie Blockoperationen oder Suchen und Ersetzen. Die Blockoperationen wurden nicht gemäß dem GEM-Standard implementiert, an den sich mittlerweile viele Programme halten. Etwas Eingewöhnung ist folglich nötig. Die gewohnte Tastaturbelegung Control+X/C/V für Cut/Copy/Paste ist nicht wiederzufinden.

Eine Tabulatorfunktion ist implementiert, sie beschränkt sich auf 8 Zeichen Breite und läßt sich nicht verändern. Ein Modula-2-Support erlaubt ferner eine Unterstützung beim Schreiben von Modula-2-Quelltexten, auch wenn dies für die Datenbank relativ unerheblich ist.

Einige Funktionen des Editors sind nicht über die Menüzeile erreichbar. Zu diesen Funktionen gehören die Spaltennummernanzeige oder auch eine Makrofunktion.

ComBase verfügt über ein kontextsensitives Manual, welches jede wichtige Unterstützung bietet. Das Manual ist sehr nützlich, wirkt jedoch etwas unfertig, da beispielsweise aus mehreren Wörtern zusammengesetzte Schlüsselbegriffe keine Leerzeichen enthalten, sondern mit Unterstrichen verbunden werden.

Integrierte Sprache

Neben dem Dialogteil von ComBase ist noch ein mächtiger Programmierteil implementiert. Mit ihm lassen sich die sogenannten Algorithmen erstellen. Der Sprachumfang deckt alles ab, was man sich von einer Programmiersprache wünscht. Dazu gehören Schleifen, bedingte Anweisungen und auch Unterprogramme.

Die Befehle gliedern sich in mehrere Gruppen: Datenbankbefehle (beispielsweise Insert, Delete, Update, First, Last), Maskenbefehle (wie Get Mask, Show Mask, New, Export), Ausgabebefehle (etwa Alert, Write(Ln), Print), Ausgabeumlenkung (Set Printer/Terminal/Serial/Textfile), Kontrollbefehle (zum Beispiel If..Else.., While..EndWhile, GoSub), Eingabebefehle (wie Answer, Askfile, Pop Selection), Umgebungsbefehle (beispielsweise Send, Rm, Exist) sowie eine Reihe von Operatoren und Funktionen.

Allgemein fällt bei ComBase auf, daß die Begriffswelt nicht mit der der relationalen Datenbanken übereinstimmt. Begriffe mit festen Bedeutungen wie Tabellen, Datenbank und Spalten werden anders benutzt. Das Relationenmodell sowie das Konzept der Regeln (beispielsweise mit kaskadiertem Löschen) über Primär- und Fremdschlüssel findet sich kaum wieder. ComBase macht einen etwas unfertigen Eindruck. Die Gestaltungsregeln für GEM-Programme sind kaum eingehalten, die Benutzung ist insgesamt zwar nicht kompliziert, dafür aber an einigen Stellen (beispielsweise bei der Auswahl von Daten) relativ umständlich.

Für die Nutzung der Datenbank über eine Programmiersprache wäre ein SQL-Interpreter oder die Möglichkeit der Nutzung von QBE (Query by Example) wünschenswert.

Die Datenbanken werden dialogorientiert angelegt. Die Benutzung der Masken ist nicht kompliziert. Die starke Aufgliederung in INIT-, REL- Mask-, List- und XPort-Modul sowie Papyrus zur Ausgabe und einen Editor zum Bearbeiten von Dateien macht eine Benutzung von ComBase etwas mühsam.

Besonders interessant ist die Möglichkeit, mit ComBase Algorithmen zu programmieren. Die leistungsfähigen Algorithmen erlauben eine vielfältige Nutzung der Datenbank und sind praktisch anzuwenden.

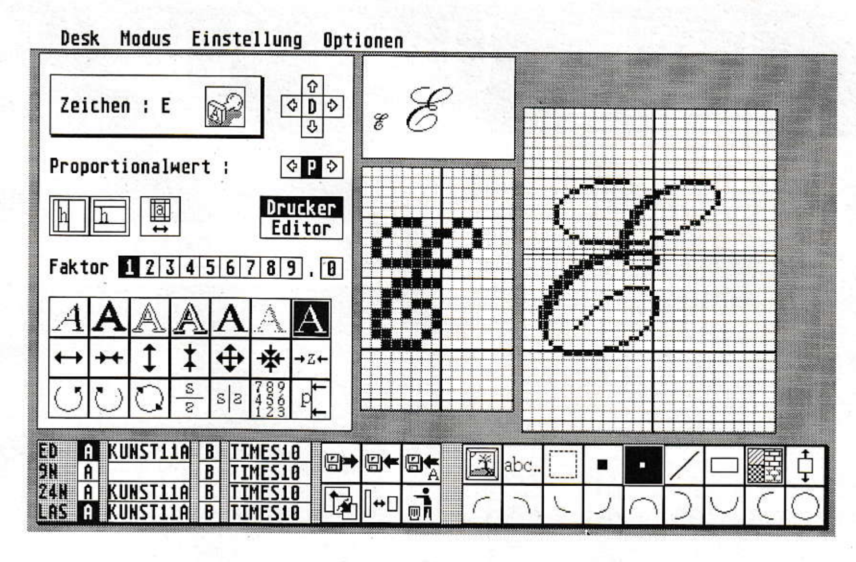
Ein Vergleich mit anderen auf dem Markt angebotenen Datenbanken ist ratsam, um vor einem Kauf entscheiden zu können, ob ComBase den gestellten Anforderungen genügt.

Dietmar Rabich

Bezugsadresse:

HEIM-Verlag
Heidelberger Landstraße 194
W-6100 Darmstadt
(06151)56057

Der professionelle Font-Editor



SCARABUS ist ein Font-Entwicklungsprogramm [Font = Zeichen(Schrift)satz] für pixelorientierte Textverarbeitungen und Grafikprogramme. In der vorliegenden Version 3.0 hat dieses universelle Werkzeug einige wichtige Verbesserungen erfahren.

SCARABUS 3.0

SCARABUS wurde nicht einem Kurztest unterzogen, sondern ich setze das Programm in der Version 3.0 schon seit Monaten ein. Insgesamt arbeite ich seit über 3 Jahren mit SCARABUS. Bis heute ist das Programm in allen seinen Versionen noch nicht ein einziges Mal abgestürzt. Das ist die Sicherheit, die ich als Entwickler benötige und schätze.

Hardware

Die Version 3.0 ist nunmehr in Turbo C geschrieben, wodurch SCARABUS nicht nur schneller, sondern auch kompakter geworden ist (40 KB kleiner trotz gestiegenen Funktionsumfanges). Ein ATARI mit 1 MB Arbeitsspeicher (Test auf ST-Book 1/40) reicht völlig aus. Voraussetzung ist lediglich ein monochromer Monitor mit der ST-hoch(640x400)-Auflösung. Ich setze SCARABUS in der Regel auf einem TT 4/16 mit Proscreen TT (19"-Großmonitor) ein. Dank des schnelleren Programmes wird die Arbeit im SM124-Modus nicht behindert.

Software-Font-Kompatibilität

Mit SCARABUS können Sie Fonts für folgende Textverarbeitungen erstellen: AKZENT II (PC-Programm), CyPress, Papyrus, Script, Signum!, Tempus Word und That´s Write. Hinzu kommen die Grafikprogramme, die das Signum!-Font-Format lesen können. Neu ist die Möglichkeit, GEM-Fonts zu laden und zu speichern. Die Größe ist dabei auf das Signum!2-Format beschränkt. Auch wenn Schriften für Signum!3 entwickelt werden sollen, empfehle ich, zuerst mit SCARABUS zu arbeiten, da das Signum!3-Format vorerst eine Sackgasse ist. Somit halten Sie sich den Weg in andere Programme offen.

Sollten sich dank schnellerer und leistungsfähigerer Rechner in der Zukunft Vektor-Fonts durchsetzen, ist SCARABUS trotzdem nicht überflüssig. Ein Pixel-Zeichen ist gegenüber Vektorzeichen schneller und einfacher erstellt. Später können Sie dann immer noch vektorisiert werden.

Wie Sie auf Bild 1 sehen, erlaubt SCARABUS, Fonts gleichzeitig für 9- und 24-Nadler (inkl. Bubblejets) sowie für Laser und 300- dpi-Tintenstrahler zu erstellen. Horizontale wie vertikale Hilfslinien nebst anzeigbarer Grundlinie und Horizontalbreite erleichtern die Orientierung. Als Werkzeuge stehen diverse Punkt-, Linien- und Kurvenformen zur Verfügung. Es können Ausschnitte definiert werden, die lösch-, verschieb- und veroderbar sind. So ist aus einem „b“ schnell ein „p“ erstellt. Zur weiteren Arbeitserleichterung sind Bilder ladbar (ab Version 3.0 auch im IMG-Format), die buchstabenweise - auch gestaucht - in die Font-Matrix übernehmbar sind.

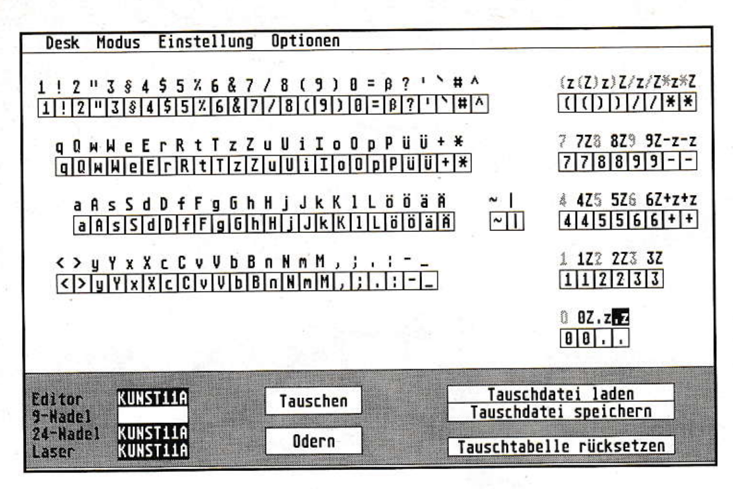
Bedienungskomfort

Neben der bereits erwähnten Stabilität und erhöhten Geschwindigkeit bietet SCARABUS 3 auch mehr Komfort und Sicherheit. Wie bereits in früheren Versionen können die wichtigsten Einstellungen abgespeichert werden. SCARABUS ist mit der Maus und - wo sinnvoll - über die Tastatur bedienbar. Wenn man wie ich manchmal 12 Stunden am Tag mit SCARABUS arbeitet, kann es vorkommen, sich beim Abspeichern im Kästchen zu verklicken. Früher überschrieb SCARABUS gnadenlos vorhandene Dateien. Jetzt erscheinen in diesem Fall Sicherheitsabfragen, die das Abbrechen ermöglichen!

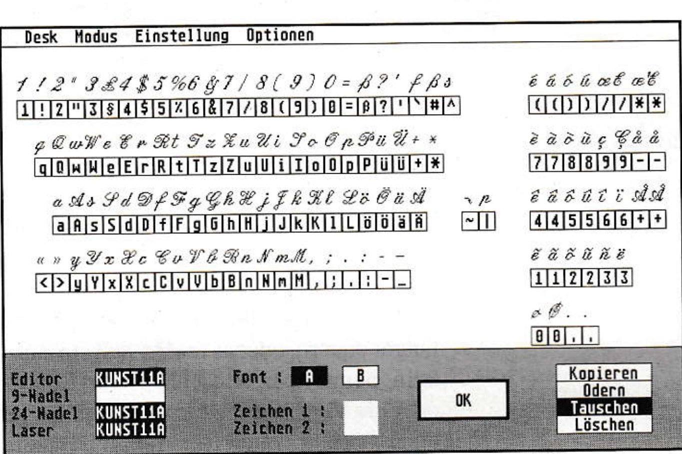
Funktionsvielfalt

Vorhandene Fonts oder Zeichen sind variabel manipulierbar: Sie können vergrößern, verkleinern, stauchen, drehen, schrägstellen, verdicken, aushöhlen, schattieren, spiegeln etc. Sofern Sie für mehrere Druckertypen gleichzeitig Fonts erstellen, können die Zeichen proportional oder 1:1 in die andere Matrix übernommen werden.

Bei außerlateinischen Schriften (z.B. Griechisch, Hebräisch, Kyrillisch) gelingt es mir nicht, eine für alle Kunden befriedigende Belegung zu erstellen. Die einen wollen eine Belegung gemäß QWERTZ (entsprechend dem Lautwert der deutschen DIN-Tastatur), die andern bevorzugen die jeweilige Landesbelegung. Andere wiederum wünschen ganz individuelle Belegungen. Dem trägt SCARABUS Rechnung mit dem Modus „Font tauschen“ (Bild 1). Über abspeicherbare Tabellen lassen sich in Sekundenschnelle individuelle Belegungen erstellen. Dabei ist es möglich, die Fonts gleichzeitig mit den Zeichen aus einem 2. Font zu versehen (natürlich alle Formate im gleichen Arbeitsgang).



Im Modus "Font tauschen" lassen sich über abspeicherbare Tabellen in Sekundenschnelle individuelle Belegungen erstellen.



Es lassen sich die Zeichensätze mit ihren wirklichen Bildschirmzeichen darstellen.

Ein weiterer Arbeitsbereich für SCARABUS ist die Erstellung von Sonderzeichen. Sofern diese wie bei Transkriptions- oder Transliterations-Fonts auf den „normalen" Fonts basieren, benötigt man die „Oder"-Funktion (odern = 2 Zeichen bzw. Ausschnitte werden miteinander verknüpft). Wollen Sie z.B. alle Vokale mit einem Akzent versehen, bietet SCARABUS 3 jetzt fürs Verodern ebenfalls die Möglichkeit, abspeicherbare Tabellen zu erstellen. Während bei früheren SCARABUS-Versionen die Proportion fast willkürlich verändert wurde, ist in der aktuellen Version dieser Fehler behoben. Stets wird der Wert des Zeichens aus Font „A" belassen.

Tauschtabellen und TOS 2.06/3.06

TOS 2.06 und 3.06 möchte ich nicht mehr missen. Im untenstehenden Fall behindern sie leider die Arbeit, was aber nicht SCARABUS anzulasten ist. Ab TOS 2.06 und 3.06 sind die Zahlenfelder des Ziffernblokkes nicht mehr über Alternate aufrufbar. Leider wird in den Tausch- und Odertabellen das „B"-Zeichen über Alternate definiert (erscheint auf dem Monitor hell - siehe Bild 1). Ich verschiebe die benötigten Zeichen daher zuerst über die Funktion „Font spezial" (Bild 3) auf das QWERTZ-Feld. Danach kann wie gewohnt gearbeitet werden.

Font spezial

Unter „Font spezial" (Bild 2) werden die Fonts (von „A" nach „B" umschaltbar) mit ihren wirklichen Bildschirmzeichen dargestellt. Neu ist, daß im aktuellen SCARABUS die volle Höhe dargestellt wird, während frühere Versionen obere und untere Bereiche der Matrix abschnitten.

Fazit

Wenn Sie für Akzent II, CyPress, Script, Signum!2, Papyrus, Tempus Word, That´s Write Zeichen verändern, umbelegen oder neue Schriften erstellen wollen, kann ich Ihnen SCARABUS voll und ganz empfehlen. Das Handbuch ist leicht verständlich und bietet alle Informationen, die zur Bedienung des Programmes nötig sind. Der Preis von DM 95,- ist für dieses professionelle, absturzsichere Werkzeug voll gerechtfertigt. Für DM 30,- (+ DM 5,- Versand) besteht die Möglichkeit zum Update beim neuen Distributor.

Veit Brixius

Bezugsadresse:

APiSoft
Bundesallee 56
W-1000 Berlin 31
(030)8534350

Änderungen gegenüber SCARABUS 2

- schneller und kürzer
- Löschen mit rechter Maustaste (einfacher/schneller)
- Sicherheitsabfrage beim Speichern
- liest/schreibt GEM-Format
- Bilder auch im IMG-Format ladbar, Ausschnitte können dabei gestaucht übernommen werden
- globales horizontales Verschieben möglich (einfaches Erstellen von Super- und Subscripten)
- diverse kleinere Fehler (u.a. beim Odern) behoben.

Monoklin

Grafische Darstellung von Molekülen

Eine der Aufgaben der Chemie ist es, neben der Erforschung neuer Synthesewege auch die Struktur, d.h. den räumlichen Aufbau, von Substanzen zu erforschen. Hierfür gibt es eine ganze Anzahl von Verfahren. Leider sind die ermittelten Daten sehr umfangreich und auch nicht besonders anschaulich, denn wer erkennt schon aus einer Datenwüste den Aufbau eines Moleküls? Dies führte zur Verwendung von Computern, die in der Lage sind, die großen Datenmengen zu verarbeiten und auch grafisch sichtbar zu machen. Nebenbei ergeben sich natürlich auch noch so nützliche Dinge wie Rotation um die Achsen, unterschiedliche Skalierungen und Darstellungen als Kalottenmodelle.

Während alle diese Anwendungen schon lange auf IBM- und VAX-Maschinen regen Gebrauch finden, war es wegen fehlender Software auf den ATARI-Computern leider nicht möglich. Einzig das PD-Programm Molekül (PD 52) bot bereits einige Möglichkeiten, war aber nicht in der Lage, die Formate aus den anderen Rechnerwelten zu verarbeiten. Hier eröffnen sich neue Möglichkeiten mit dem Programm Monoklin, welches sowohl Molekül-Dateien, zusätzlich aber auch die Formate der Marktführer SCHAKAL und ALCHEMY sowie SHELXTL verarbeiten kann. Ebenso eingeschlossen ist eine Konvertierung in obige Formate. Neben diesen Funktionen bietet Monoklin in der Version 1.1 aber noch eine Menge mehr hilfreiche Dienste an.

Bearbeitungsmöglichkeiten

So kann die Atomliste der Dateien komfortabel editiert und verändert werden. Es sind zwei Varianten der Bindungsberechnung möglich, und es können zusätzlich weitere Bindungen eingefügt bzw. gelöscht werden. Wenn nötig, läßt sich auch das Molekül spiegeln und zentrieren und durch Definition von Ebenen und Geraden in eine konkrete Ansicht drehen.

Für Syntheseschritte ist es oft entscheidend, die Winkel und Abstände der Atome zu kennen. Monoklin berechnet diese wahlweise durch Atomnummerneingabe oder besonders komfortabel durch Anwahl mit der Maus. Auf Wunsch kann die Molekül- und Bindungsliste auf dem Drucker ausgegeben werden.

Grafische Darstellung

Der aufwendigste Teil von Monoklin ist der Grafikteil. Nicht nur, daß die Bindungslisten grafisch dargestellt werden, auch eine umfangreiche Nachbearbeitung ist möglich. Im Rotationsteil kann das Molekül um alle drei Achsen gedreht und seine Lage verändert und gezoomt werden. Mit einem 8-MHz-Rechner kann bei größeren Molekülen die Rotation etwas länger dauern, oder es macht sich ein Rukkeln bemerkbar. Beschleuniger-Boards oder die Verwendung eines TTs führen zu einer störungsfreien Rotation in Echtzeit. Interessant ist es, vom Molekül durch Spiegelung ein Enantionmer zu erhalten, welches dann neben dem Original dargestellt wird. Auch die Projektion läßt sich durch Parameterveränderung den Darstellungswünschen anpassen. Sind alle Einstellungen nach Wunsch gemacht worden, kann durch die Vollbildoption das Molekül in einer kalottenmodellähnlichen Darstellung gezeichnet werden. Dazu kann jedem Atom ein eigenes Füllmuster zugeordnet werden. Durch Verwendung einer Lichtquelle und einer Outline-Darstellung der Bindungen sind Aufwertungen des Bildes möglich. Um eine maximale Ausgabequa-

Messen
Winkel W
Torsionswinkel T
Abstände A
per Mausklick M

Drucken
Atomliste
Protokoll
Hardcopy
Plotter
Parameter

Grafik
rotieren R
Vollbild V
Optionen O
Neu zeichnen N

Hilfe
ein/aus

Utility
CHN-Analyse F1
Assign F2
List File F3
lade INF F4
Molpeak F5
Execute F6
LST->SHK F7
nicht belegt
nicht belegt
nicht belegt

MONOKLIN
Prg Info

File
laden ^L
speichern ^S
umbenennen ^U
kopieren ^C
löschen ^D
editieren ^E
QUIT ^Q

Arbeiten
Bindungen
berechnen B
einfügen E
löschen L
spiegeln S
zentrieren Z
Ursprung festlegen U
Ebene festlegen P
Gerade festlegen G

Die Pull-down-Menüs von Monoklin

That's Write 2 398,- DM*

Das Textsystem der Superlativen !!

Was ist That's Write?
Mit That's Write steht Ihnen ein umfangreiches Textprogramm für Ihren Atari TOS Computer zur Verfügung. Durch die leichte Bedienung per Maus ist das Programm für den Einsteiger schnell erlernbar. Für den Vielschreiber bietet es zusätzliche Bedienung über die Tastatur. Tastenbelegung für Standard- und eigene Funktionen können über Makro Definitionen realisiert werden. That's Write arbeitet wie viele DTP-Programme mit Absatz- und Seitenlayouts. Die Bildschirmanzeige entspricht bereits dem Ausdruck (WYSIWYG).

Wer soll That's Write kaufen?
Einsteiger, Vielschreiber und Schnellschreiber, jeder der eine perfekte Textverarbeitung braucht. Texterfassung, Büro, Uni, Privat...

Im Büro
Textbausteine, Floskeln, Makros, Rechnen im Text, Serienbrief usw. ermöglichen das einfache Erstellen von Einladungen, Werbebriefen, Angeboten, Rechnungen, Kundeninformationen...

An der Uni
Facharbeiten leicht und gut gemacht mit Hilfe von Gliederungsfunktion, Fußnoten- und gleichzeitiger Endnotenverwaltung, automatischem Erstellen von Stichwort-, Inhalts-, Abbildungs-, Tabellenverzeichnis...

Zu Hause
Vom einfachen Brief bis zum Buch mit 999 Seiten mit automatischer Kapitel-, Bilder-, ...numerierung, Silbentrennung und Korrektur, mit Einbindung von Grafiken im IMG-Format

Unterstütze Drucker
Atari SLM 804 / 605 / ... Laserdrucker
Epson, Star, Nec, Panasonic, Oki, Seikosha und kompatible 9 / 24-Nadeldrucker
Canon BJ, HP Deskjet, Deskjet Plus und Deskjet 500 Tintenstrahldrucker
HP Laserjet, EPSON GQ / EPL und dazu kompatible Laserdrucker

That's Address 198,- DM*

Auf gute Zusammenarbeit... (z.B. mit That's Write:)
Die schnelle und bequeme Lösung, um beliebige Adressen gleichzeitig zu verwalten, Listen und Serienbriefe zu erstellen. That's Address arbeitet als Accessory (daher direkt in That's Write verfügbar). Ein Tastendruck und die gewünschte Adresse erscheint an gewünschter Stelle in Ihrem Brief.
Kartei: Im Gegensatz zu einer herkömmlichen Kartei können Sie in That's Address zu einer Adresse nahezu beliebig viele weitere Daten nach Ihrem Bedarf erfassen.
Datensicher: Eingegebene oder geänderte Daten (und nur diese) werden sofort auf Festplatte gespeichert - schnell und sicher.
Telefon: Wahl mit geeignetem Modem. Durch Angabe des eigenen Standortes wird im gleichen Ortsnetz die Vorwahl automatisch unterdrückt.
Hilfreich: '5 21' wird '5000 Köln 21' umfangreiche Ortedatei inklusive International: Auch 'Huntingdon PE17 5JA' ist kein Problem. Das Programm sucht sich die Postleitzahl selbst anhand einer Länderschablone.

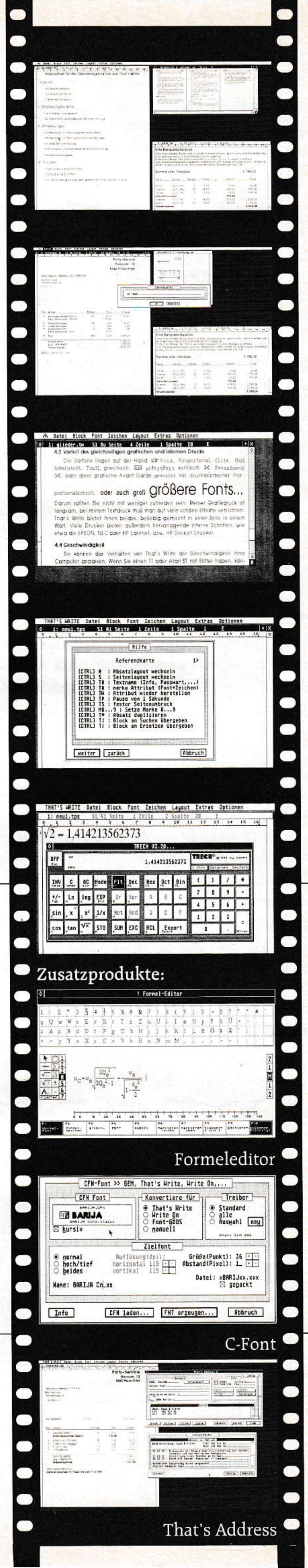
*unverbindliche Preisempfehlung

Compo Software ist Spitzenqualität – erhältlich bei guten Fachhändlern

Bezugsquellen finden Sie auf der übernächsten Seite

Hotline + Info:
Compo Software
Tel. 06551-6266
Fax 06551-6339

Vertrieb + Info: ***Heim Verlag***
Heidelb. Landstr. 194
6100 Da.-Eberstadt
Tel. 06151-56057/8
Fax 06151-56059



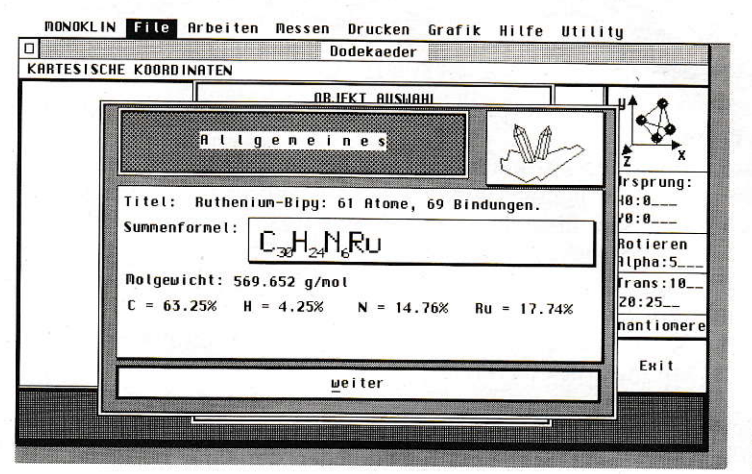
lität beim Druck zu erreichen, druckt Monoklin das Molekül über GDOS aus. Eine Ausgabe als Metafile ist ebenfalls möglich.

Utilities

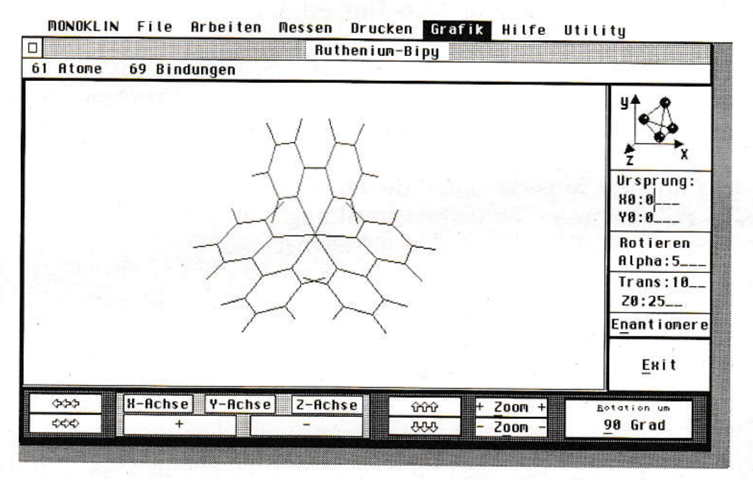
Hier sind einige nützliche Zusatzroutinen zusammengefaßt. Für Elementaranalysen ist eine CHN-Analyse eingebaut, die nach Eingabe der Summenformel die Zusammensetzung in Prozent der Elemente ermittelt. Dazu kommen Routinen für die Umbenennung der Atom-Labels und Filelistings. Sehr gut ist auch, den Mol-Peak seines Moleküls berechnen zu lassen und dann mit den Meßwerten vergleichen zu können.

Fehler und Inkompatibilitäten

Während des Tests lief Monoklin sehr stabil. Leider zeigte sich, daß das Programm durch Verwendung von GFA-BASIC noch LINE-A-Zugriffe macht und einige andere Unsauberkeiten enthält, die auf die Programmiersprache zurückzuführen sind. Dies führt leider dazu, daß Monoklin auf Grafikkarten nicht lauffähig ist und mit zu großen Grafikspeichern Probleme bekommt. Lauffähig ist es mit Overscan und anderen auf dem ST erhältlichen Grafikerweiterungen (Pixel Wonder usw.) auf dem TT und mit Beschleuniger-Boards. Weiterhin konnten leider nicht alle Programme die von Monoklin erstellten Metafiles lesen. Im Notfall kann eine Konvertierung mittels Easydraw durchgeführt werden. Warum die Metafiles nicht von allen Programmen verarbeitet werden konnten, ließ sich leider nicht feststellen.



Einladen einer Datei



Im Rotationsteil kann das Molekül um alle drei Achsen gedreht und seine Lage verändert und gezoomt werden.

Zusammenfassung

Monoklin stellt ein mächtiges Werkzeug dar. Es ist sinnvoll für alle Chemiker, die Moleküle darstellen und bearbeiten möchten und bisher immer die Programme anderer Systeme verwenden mußten. Der geringe Preis von 40,- DM als MAXON-Sonderdisk (Nr. 80) und die Tatsache, daß Monoklin zur Zeit konkurrenzlos ist, lassen die oben angeführten kleinen Schwächen verschmerzen. Ich denke, daß diese Probleme demnächst mit einem Update gelöst werden können. Als kleine Erweiterung wäre noch ein Animationsteil denkbar, mit dem Vollbildausgaben gedreht werden können.

Rainer Esser

Bezugsquelle:

MAXON-Computer
Schwalbacher Straße 52
W-6236 Eschborn
Tel.:(06196) 481811









CompoScript

598,- DM*

*unverbindliche Preisempfehlung

PostScript auf Ihrem Atari TOS Computer und Ihrem Drucker !!

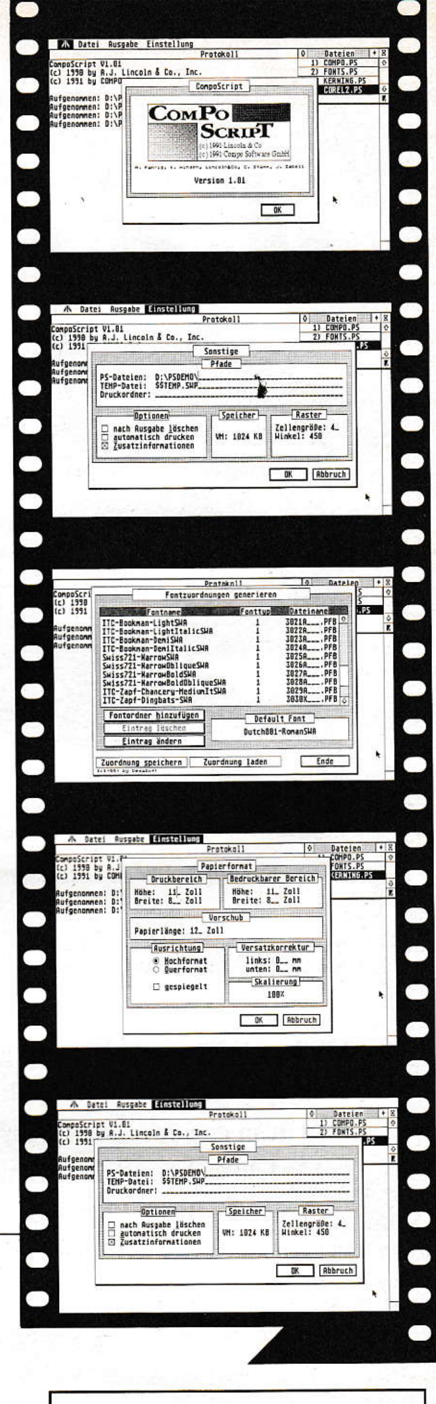
Was ist CompoScript?
CompoScript ist ein PostScript Interpreter für Ihren Atari TOS Computer. Mit diesem können Sie PostScript Dateien auf nahezu jedem Drucker drucken. Sie können diese auch wahlweise auf den Bildschirm ausgeben oder in eine IMG oder TIFF Grafikdatei konvertieren.

Brauchen Sie CompoScript?
Wenn Sie DTP, Text, CAD, Grafik, Notensatz oder Präsentationssoftware einsetzen, auf Ihrem Atari, einem PC-Emulator oder von sonst einem Betriebssystem Ihre Daten erhalten, so können Sie sicherlich von CompoScript profitieren. Viele Programme können PostScript Dateien erzeugen. Diese können auch Sie nun in bestmöglicher Qualität auf Bildschirm und Drucker ausgeben. CompoScript setzt auch Ihre PostScript Clip-Art Grafikdateien in IMG oder TIFF um. Oder lassen Sie sich eine PS-Datei in ein Fenster ausgeben und sehen sich diese in der gewünschten Auflösung an, bevor Sie sie ausdrucken.

35 Fonts
CompoScript kommt gleich mit den 35 hochwertigen Standard Type 1 Fonts der Firma Bitstream. Diese verfügen selbstverständlich über 'Hints'. Dadurch kann selbst bei niedrigen Auflösungen höchste Kompatibilität zu anderen Geräten mit PostScript-Ausgabe erreicht werden.

Unterstützte Drucker
Atari SLM 804 / 605 / ... Laserdrucker
Epson, Star, Nec, Panasonic, Oki, Seikosha und kompatible 9 / 24-Nadeldrucker
Canon BJ, HP Deskjet, Deskjet Plus und Deskjet 500 Tintenstrahldrucker
HP Laserjet, EPSON GQ / EPL und dazu kompatible Laserdrucker

Systemanforderung
Atari TOS Computer mit wenigstens 1Mb Arbeitsspeicher (4Mb für Atari SLM Laserdrucker) und einer Festplatte.



Dataplay
Bundesallee 25
1000 Berlin 31
Tel. 030-8619161

alpha comp. g.m.b.h.
Kurfürstendamm 121 a
1000 Berlin 31
Tel. 030-8911082

tri-tec Computer
Rigaer Str. 2
1034 Berlin
Tel. 030-5891928

Sienknecht
Bürokommunikation
Heiligengeiststr. 20
2120 Lüneburg
Tel. 04131-46122

Der Computerladen
Coriansberg 2
2210 Itzehoe
Tel. 04821-3390

Computer + Service
DTP-Center
Gutenbergstr. 2
2300 Kiel 1
Tel. 0431-569444

WBW-Service
Willi B. Werk
Sielwall 87
2800 Bremen 1
Tel. 0421-75116

PS-DATA
Faulenstr. 48 - 52
2800 Bremen 1
Tel. 0421-170577

Chemo-Soft
Lindenhofsgarten 1
2900 Oldenburg 17
Tel. 0441-82851

T.U.M. Soft- & Hardw.
Hauptstr. 67
2905 Edewecht
Tel. 04405-6809

REITMAIER
MUSIKELEKTRONIK
Parkstr. 9
3501 Habichtswald
Tel. 05606-9812

W. Wohlfahrtstätter
Hard- und Software
Irenenstr. 76 C
4000 Düsseldorf
Tel. 0211-429876

M C S
Midi & Comp. Systeme
Baroper Bhf. Str. 53
4600 Dortmund 50
Tel. 0231-759283

Meyer & Jacob
Münsterstraße 141
4600 Dortmund
Tel. 0231-833205

cc Comp. Studio GmbH
Elisabethstr. 5
4600 Dortmund 1
Tel. 0231-528184

C S A
Computersyst. u. Anw.
Wilhelminenstr. 29
4650 Gelsenkirchen
Tel. 0209-42011

AB -COMPUTER
Mommsenstr. 72
5000 Köln 41
Tel. 0221-4301442

Rolf Rocke
Comp.-Fachgeschäft
Auestraße 1
5090 Leverkusen
Tel. 02171-2624

foxware Comp. GmbH
Kurze Straße 1
5600 Wuppertal 22
Tel. 0202-640389

Mega Team
Comp. Vertriebs oHG
Rathausstr. 1 - 3
5650 Solingen 1
Tel. 0212-45888

Axel Böckem
Comp. + Textsysteme
Eilper Str. 60
5800 Hagen
Tel. 02331-73490

Eickmann Computer
I.d.Römerstadt 249/253
6000 Frankfurt M. 90
Tel. 069-763409

Heim GmbH
Büro- u. Computertechn.
Heidelb. Landstr. 194
6100 DA-Eberstadt
Tel. 06151-56057/8

K F C
Computersysteme
Wiesenstr. 18
6240 Königstein
Tel. 06174-3033

Etzkorn Computer
Austr. 20
6720 Speyer
Tel. 06232-32435

Walliser + Co.
Personal Computer
Marktstr. 48
7000 Stuttgart-Bd-Cst.
Tel. 0711-567143

Akzente
Softwarevertrieb
Schlehenweg 12
7080 Aalen
Tel. 07361-36606

Weeske
D. Elektroh. am Nordr.
Potsdamer Ring 10
7150 Backnang
Tel. 07191-1528

DON'T PANIC
Comp.-Komplettlös.GmbH
Pfleghofstr. 3
7400 Tübingen
Tel. 07071-92880

Duffner Computer
Waldkircherstr. 61 - 63
7800 Freiburg
Tel. 0761-51555-0

COMTEX
COMPUTERSYSTEME
Gitteweg 3
7801 Bollschweil
Tel. 07633-50784

schulz-computer
Landwehrstr. 35
8000 München 2
Tel. 089-597330/39

SOFTHANSA
SOFT- U. HARDWARE
Untersbergstr. 22
8000 München-Giesing
Tel. 089-6972206

HIB Computer GmbH
Äuß. Bayreuth. Str. 57-59
8500 Nürnberg 10

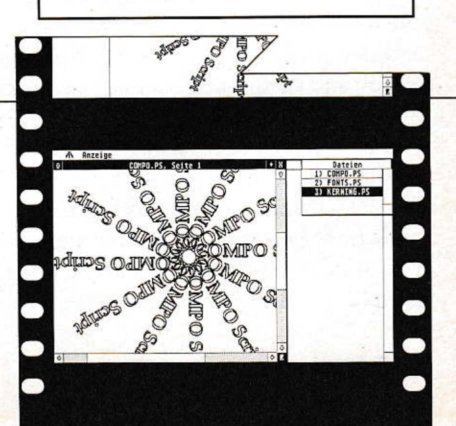
SCHÖLL BÜROTEAM
computer-center
am Dominikanerplatz 5
8700 Würzburg
Tel. 0931-30808-0

Adolf & Schmoll
Computer
Schwalbenstr. 1
8900 Augsburg-Pfersee
Tel. 0821-528533

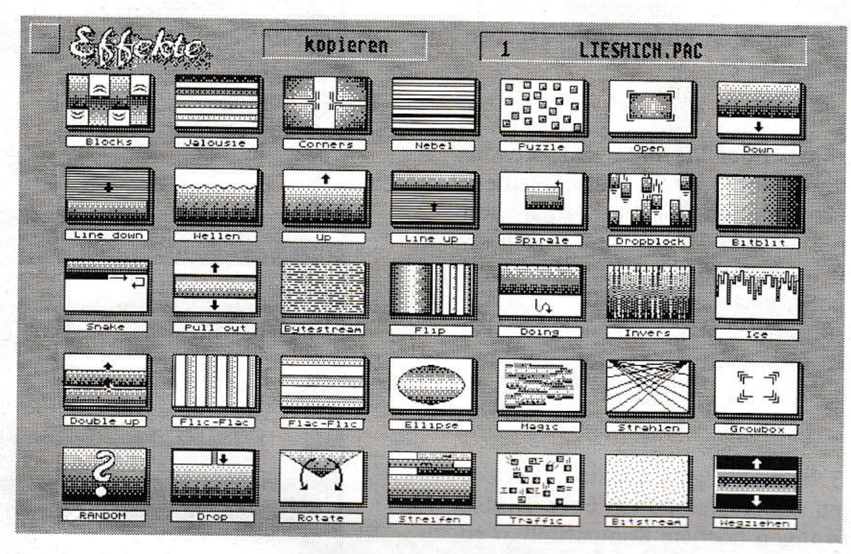
Compo Software ist Spitzenqualität – erhältlich bei guten Fachhändlern

Vertrieb + Info:
Heim Verlag
Tel. 06151-56057/8
Fax 06151-56059

Hotline + Info:
Compo Software
Tel. 06551-6266, Fax 6339



MagicView



35 Überblendeffekte für die Präsentationsfunktion

Magische Bilderwelt

Bildbearbeitungsprogramme gibt es schon genügend, MagicView legt Wert auf darauf, ein Bild*VER*arbeitungsprogramm zu sein. Was soll man denn darunter verstehen? Im Grund möchte MagicView professional V:1.3 (so der volle Name) nur fertige Bilder in den Formaten PIC (Screen 32 KByte), PI3 (Degas mono), PAC (STAD), CPG (Printing Press) oder IMG (GEM-Image, 640 * 400) verarbeiten. An irgendwelche Manipulationsmöglichkeiten, wie das Mal- und Retoucheprogramme bieten, ist dabei nicht gedacht.

Vielmehr ist der Hauptdaseinszweck das Zusammenstellen sogenannter „Slide-Shows", also ein Bilderabrollen z.B. für eine Messepräsentation. Dabei ist das Darstellungsformat auf die hohe ST-Auflösung (640 * 400 Bildpunkte) beschränkt. Ein kleines Zusatzprogramm namens „MagicShow" läßt mittlerweile auch die niedere ST-Auflösung zu.

Die Dateien holt sich MagicView immer von einem logischen Medium, also von einer Diskette oder einer Festplattenpartition. Im letzteren Fall kann man wählen, ob alle Partitionen oder nur eine nach Bildern durchsucht werden sollen. Nur wenn man eine Partition auswählt, gibt es die zusätzliche Abfrage, ob vielleicht nur ein Verzeichnis die Bilder beherbergt.

MagicView greift die Bilder auch immer in ihrer physikalischen Reihenfolge ab, d.h. das Bild, welches zuerst auf das Medium gespeichert wurde, wird auch als erstes in der Slide-Show angezeigt. Eine individuelle Reihenfolge, vielleicht auch mit Wiederholungen, gibt es leider nicht.

Showtime

Die Anzahl von 35 Überblendeffekte (s. Bild 1) kann sich sehen lassen. Das scheint mir auch die Stärke dieses Programms zu sein. Beim kurzen Test aller Effekte kamen mir einige denn doch gähnend lang vor - vielleicht könnte man eine Zeitbeschränkung oder Geschwindigkeitsvorgaben für die Effekte einbauen? Leider gibt es keine Möglichkeit, einen langwierigen Überblender per Tastendruck abzubrechen oder abzukürzen.
In einer Parameterbox kann man neben dem relevanten Laufwerk auch die Dateierweiterungen aussuchen sowie die Pausenzeiten zwischen den Bildwechseln (natürlich nur für den automatischen Bildwechsel interessant).

Beiwerk

Neben dem Laden von Bilddateien gibt es noch die Möglichkeit, diese in einem anderen Format wieder abzuspeichern. Insofern ist MagicView auch ein Grafik-Konverter. Auch Optionen zum Ausdruck (1:1, 8 Stück verkleinert, gedreht) gibt es. Weitere Utilities sind eine Formatierfunktion und die Abfrage nach dem Füllgrad eines Speichermediums.Ursprünglich als PD, später als Shareware und nun als Kommerzprodukt, hat sich MagicView ganz gut gemausert. Dennoch hat es noch einen gewissen Weg vor sich, bis es als „runde Sache" betrachtet werden kann. Irgendwie werde ich das Gefühl nicht los, daß die Formatier- und Partitionsinfo-Utilities hier fehl am Platz sind. Vielmehr hätte sich der Programmierer um ein paar mehr einlesbare Grafikformate kümmern sollen (was er vielleicht noch tun wird?). Platz für weitere Funktionen ist ja auch noch reichlich, wenn nicht in der Parameterbox, so doch in den etwas unterrepräsentierten Menüpunkten.

Das Handbuch mit seinen 19 Seiten hat einen erstaunlich hohen Bilderdurchsatz und reicht in Umfang und Sprachstil völlig zum Verständnis des Programms aus. MagicView professional kostet in der Version 1.3 129,- DM. Übrigens: Ein Upgrade von älteren Versionen kostet 100,- DM, wobei alle bisher geleisteten Registriergebühren verrechnet werden.

Leider kenne ich nun schon die Version 1.0, die noch 65,- DM kostete, und muß bemerken, daß die starke Überarbeitung von V:1.0 nach V:1.3 und das neue Zusatzprogramm für ST-niedrige Auflösung zwar einen höheren Preis rechtfertigen, nicht aber eine Verdoppelung desselben. Auf dem Weg zu einem kompletten („runden") Kommerzprodukt sind wohl noch ein paar kleine Korrekturen nötig (vielleicht wären Großbildschirmfähigkeit und größere Dateiformate machbar), vor allem aber eine große im Preis.

DK

Bezugsquelle:

Delta Labs Software
Bernhard Artz
Rembrandtstraße 1
W-5600 Wuppertal 11

Programmier praxis

STOP dem Cartridge-Klau

Dem Diebstahl ei nes Wechselplattenmediums kann sehr wirkungsvoll vorgebeugt werden. Eine Wechselplatte erlaubt es nämlich, den Auswurfknopf per Software zu verriegeln. In diesem Fall ist ein Entnehmen der Cartridge unmöglich. Wird der Auswurfknopf betätigt, läßt sich die Platte in keiner Weise beeinflussen.

Mischen und Sortieren

Computer sind besonders zum Sortieren von Daten geeignet, aber was ist, wenn man das Gegenteil vor hat; wenn man Daten durcheinanderbringen will? Auch hierzu gibt es Lösungen, eine davon sind Zufallszahlen.

Blockorientiertes Bildpacken

Die meisten Packformate für S/W-Bilder packen aufeinanderfolgende Bytes oder Byte-Folgen. Dies geht schnell und ist ist recht einfach, jedoch nicht für alle Fälle optimal. Das hier vorgestellte Verfahren packt nicht Byte-Folgen, sondern rechteckige Byte-Blöcke und komprimiert so einen 32-KByte-Bildschirm.

AT
Watchtower im Atari

In der UNIX-Welt ist die Hintergrundverarbeitung von Kommandos gang und gebe. Einen wichtigen Bestandteil zur Büroautomatisation bietet der Befehl AT. Auf ihn bauen Mail-Systeme, Backup-Lösungen und andere komplexe Büroaufgaben auf.

Text-Hardcopy
Die etwas andere Art der Hardcopy

Das Hardcopy-Problem auf dem Atari ST ist eigentlich durch eine große Anzahl von Hardcopy-Utilities ausreichend gelöst. Man kann Teile des Bildschirms (bzw. den ganzen) als Grafik ausdrucken bzw. auf Diskette speichern. Das hier vorgestellte Programm verfolgt einen anderen Zweck, nämlich eine Hardcopy eines Textes zu machen und ihn in ASCII-Zeichen abzuspeichern.

Programmierpraxis

STOP dem Cartridge-Klau

Auch auf der diesjährigen Atari-Messe werden wieder viele Anbieter mit Wechselplatten ausgestattet sein. Kein Wunder, denn so lassen sich auch größere Datenmengen leicht transportieren, und die Cartridge kann nach dem Ende einer Demonstration an einem sicheren Ort aufbewahrt werden. Was aber, wenn ein Unbefugter mal schnell den Auswurfknopf betätigt und die Cartridge mitgehen läßt?

Uwe Seimet

Diese Situation ist sicherlich für viele ein Alptraum. Selbst wenn sich keine wirklich wichtigen Daten auf dem Medium befinden (Backups sollten anderweitig vorhanden sein), ist der Cartridge-Verlust zumindest aus finanzieller Sicht ärgerlich.em Diebstahl eines Mediums kann jedoch sehr wirkungsvoll vorgebeugt werden. Eine Wechselplatte erlaubt es nämlich, den Auswurfknopf per Software zu verriegeln. In diesem Fall ist ein Entnehmen der Cartridge unmöglich. Wird der Auswurfknopf betätigt, läßt sich die Platte in keiner Weise beeinflussen. Verspürt man irgendwann den Wunsch, die Cartridge zu entnehmen, läßt sich der Auswurfknopf natürlich auch wieder entriegeln.

Besitzt man einen Festplattentreiber, der die XHDI-Spezifikation (s. hierzu „Schoner für die Festplatte“, ebenfalls in dieser Ausgabe) erfüllt, so läßt sich dieses Vorhaben leicht realisieren. Der XHDI-Aufruf *XHLock()* erlaubt das treiberunabhängige Ver- und Entriegeln des Auswurfknopfes. Nachdem mit *XHInqTarget()* festgestellt wurde, bei welcher Platte es sich um eine Wechselplatte handelt, lassen sich alle Cartridges verriegeln. Hierzu dient LOCK.PRG, wogegen UNLOCK.PRG die Verriegelung wieder aufhebt.

Da sich beide Programme lediglich durch einen Parameter beim*XHLock()*-Aufruf unterscheiden, gibt es nur ein Programm-Listing. Je nachdem, welche Programmversion benötigt wird, muß einer der beiden MODE-Definitionen am Anfang des Listings auskommentiert werden. Das Programm überprüft übrigens nicht, ob sich zwei logische Laufwerke auf dem gleichen physikalischen Device befinden, so daß es durchaus vorkommen kann, daß der *XHLock()* für eine Platte mehrmals abgesetzt wird. Dies hat jedoch keine nachteilige Wirkung.

```
******************************************
*                                        *
*             LOCK/UNLOCK V1.00          *
*                                        *
* Ver- oder entriegelt den Auswurfknopf  *
*  von Wechselplatten unter Verwendung   *
*          des XHDI-Protokolls           *
*                                        *
*              by Uwe Seimet             *
*                                        *
*       (c) 1992 MAXON Computer          *
*                                        *
******************************************


*MODE-Parameter je nach Bedarf auskommentieren
*MODE     = 0       ;Cartridge freigeben
MODE      = 1       ;Cartridge verriegeln


GEMDOS    = 1

XBIOS     = 14
SUPEXEC   = 38

_p_cookies = $5a0

XHInqTarget = 1
XHLock      = 3
XHDrvMap    = 6
XHInqDev    = 7

XHDIMAGIC = $27011992


          text

          pea act(pc)
          move #SUPEXEC,-(sp)
          trap #XBIOS
          addq.l #6,sp
          clr -(sp)
          trap #GEMDOS

act:
          move.l _p_cookies,d0         ;keine
          beq quit                     ;cookies-
          move.l d0,a0
noxhdi:   movem.l (a0)+,d0/a6
          tst.l d0                     ;kein
          beq quit                     ;XNDI-
          cmp.l #"XHDI",d0
          bne noxhdi
          cmp.l #XHDIMAGIC,-4(a6)
          bne noxhdi

          move #XHDrvMap,-(sp)         ;Gerätevektor
          jsr (a6)                     ;holen
          addq.l #2,sp
          move.l d0,d6

          moveq #0,d7
loop:     btst d6,d7                   ;kein XHDI
          beq.s noxdev                 ;Device-

          clr.l -(sp)
          clr.l -(sp)
          pea minor
          pea major
          move d7,-(sp)
          move #XHInqDev,-(sp)         ;Gerätenummern
          jsr (a6)                     ;holen
          lea 20(sp),sp
          tst d0
          bmi.s noxdev

          clr.l -(sp)
          pea flags
          clr.l -(sp)
          move minor,-(sp)
```

→




```
         move major,-(sp)
         move #XHInqTarget,-(sp)
         jsr (a6)
         lea 18(sp),sp
         tst d0
         bmi.s noxdev
         btst #2,flags+3          ;kann Gerät
                                  ;verriegelt
                                  ;werden?
         beq.s noxdev             ;nein-
         tst.l flags              ;Gerät ist
         bmi.s noxdev             ;reserviert-

         clr -(sp)                ;Dummy-Key
         move #MODE,-(sp)         ;Cartridge
         move minor,-(sp)         ;sperren oder
         move major,-(sp)         ;entriegel
         move #XHLock,-(sp)       (gemäß MODE)
         jsr (a6)
         lea 10(sp),sp

noxdev:  addq #1,d7
         cmp #32,d7               ;nächstes
         bne loop                 ;Device-

quit:    clr -(sp)
         trap #GEMDOS


         bss

flags:   ds.l 1

major:   ds.w 1

minor:   ds.w 1
```

Mischen und Sortieren

Computer sind besonders zum Sortieren von Daten geeignet, aber was ist, wenn man das Gegenteil vor hat; wenn man Daten durcheinanderbringen will? Auch hierzu gibt es Lösungen, eine davon sind Zufallszahlen. Man kann sie gut gebrauchen, um willkürliche Beispieldaten für ein Programm zu erstellen. Auch in Spielen sind Zufallszahlen sehr wichtig.

Arnd Rosemeier

Ein guter Zufallszahlengenerator sollte eine ungefähre Gleichverteilung seiner Ergebnisse liefern. Bei dem ersten Programm handelt es sich um ein Testprogramm für den Zufallszahlengenerator von Omikron.BASIC. Die beiden obersten Zeilen des Programms dienen der Einstellung Ihrer Bildschirmauflösung. Wenn Sie mit dem monochromen Monitor SM-124 von Atari arbeiten, dann lassen Sie die Einstellungen so, wie sie sind. Die Variable Step enthält die horizontale Auflösung des Tests. Kleinere Zahlen bringen hier eine höhere Auflösung. Scale enthält die vertikale Auflösung. Auch hier gilt: je kleiner, desto genauer. Die FOR-Schleife am Anfang dient der Erzeugung von Zufallszahlen. Wenn Sie einen selbstgeschriebenen Zufallsgenerator testen wollen, dann müssen Sie hier den RND-Befehl durch Ihren eigenen Unterprogrammaufruf ersetzen. Der Befehl TEXT HEIGHT = 4 schaltet für den TEXT-Befehl auf den 6x6-Systemzeichensatz um. An dieser Stelle ist auch mal ein bißchen Aufklärungsarbeit in bezug auf den TEXT-Befehl angebracht. Die y-Koordinate in diesem Befehl gibt nicht etwa die obere Ecke für die Textausgabe an, sondern die Stelle, an der die Grundlinie des Textes steht. Die Grundlinie ist beim 6x6-Zeichensatz vier Punkte unterhalb der oberen Ecke. Dies ist auch der Grund, weshalb TEXT HEIGHT = 4 auf den 6x6-Zeichensatz umschaltet.

Das zweite Programm ist für alle Kartenspieler interessant. In diesem Programm wird ein sortiertes Kartenspiel gemischt und danach wieder sortiert. Die Karten werden dabei durch Zahlen von 0 bis 31 repräsentiert. Die Farbe einer Karte kann man mit INT(Karte/8) herausbekommen. Man erhält einen Wert von 0 bis 3, den man sehr einfach einer Zeichenkette, z.B. „Karo" zuordnen kann. Die Karte selber erhält man durch (Karte MOD 8), also dem Divisionsrest. Etwas schneller und auch ‚computermäßiger' wäre der Ausdruck (Karte AND 7), der zufälligerweise hier das gleiche Ergebnis liefern würde, der dafür aber schon das Verständnis von bitweisen Verknüpfungen vorraussetzen würde. Interressant ist vielleicht auch die Methode, wie man ein Kartenspiel mischen kann. Dazu wird einfach jede Karte mit einer beliebigen, zufälligen anderen Karte vertauscht. Auf diese Weise dürfte eine bessere Durchmischung gewährleistet sein, als man sie durch manuelles Mischen erreichen könnte.

```
' (c)1992 by MAXON-Computer
Xmax%L=640
Ymax%L=400
Sstep%L=20
Scale%L=1
Mhoehe%L=Ymax%L/Scale%L/2
X1%L= INT(Xmax%L/Sstep%L)
DIM A%L(X1%L)
PRINT "Ich berechne Zufallszahlen..."
FOR I%L=0 TO X1%L*Mhoehe%L-1
  Zufall%L= RND(X1%L+1)
  A%L(Zufall%L)=A%L(Zufall%L)+1
  IF (I%L AND 511)=511 THEN PRINT ".";
NEXT I%L
PRINT
CLS : PRINT CHR$(27);"f";: MODE =0: TEXT HEIGHT =4
FILL COLOR =1: FILL STYLE =2,4: OUTLINE ON : CLIP 0,0,Xmax%L,Ymax%L
FOR I%L=0 TO X1%L
  PBOX I%L*Sstep%L,Ymax%L-A%L(I%L)*Scale%L TO (I%L+1)*Sstep%L,Ymax%L-1
  IF Sstep%L>=14 THEN
    TEXT I%L*Sstep%L+(Sstep%L-12)/2,Ymax%L-5, RIGHT$(STR$(100+I%L),2)
    A$= MID$( STR$(A%L(I%L)),2):A1%L= LEN(A$)*6
    TEXT I%L*Sstep%L+(Sstep%L-A1%L)/2,Ymax%L-A%L(I%L)*Scale%L-2,A$
  ENDIF
NEXT I%L
```

```
' (c)1992 by MAXON-Computer
DIM A%L(31),Farbe$(3),Karte$(7)
FOR I%L=0 TO 3: READ Farbe$(I%L): NEXT I%L
FOR I%L=0 TO 7: READ Karte$(I%L): NEXT I%L
DATA "Karo","Herz","Pik","Kreuz"
DATA "Sieben","Acht","Neun","Zehn","Bube","Dame","König","Ass"
FOR I%L=0 TO 31
  A%L(I%L)=I%L
NEXT I%L
' ***Jetzt mischen!
FOR I%L=0 TO 31
  SWAP A%L(I%L),A%L( RND(32))
NEXT I%L
' ***Gemischt ausgeben
Ausgabe
' ***Jetzt wieder sortieren
SORT A%L(0)
' ***und dann ausgeben
Ausgabe
END
'
'***** Prozeduren *****
'
DEF PROC Ausgabe
LOCAL I%L
  ' ***Automat. Zeilenumbruch einschalten (BASIC ab Version 3.5)
  PRINT CHR$(27);"v";
  FOR I%L=0 TO 31
    IF I%L THEN PRINT ", ";
    PRINT Farbe$( INT(A%L(I%L)/8));" "; Karte$(A%L(I%L) MOD 8);
  NEXT I%L
  PRINT
  PRINT
RETURN
```

Software

Papyrus	269,--
Calamus 1.09N	328,--
Signum III	458,--
NEU CyPress	**308,--**
Annabel RCS	109,--
Interface RCS	89,--
Argon Backup	95,--
Mag!X	128,--
FastCopy Pro	85,--
Kobold	75,--
Mortimer plus	119,--
Poison Virenschutz	**86,--**
Data Diet	123,--
BTX ST online plus	192,--
Megapaint 4.0 Book	198,--
Arabesque pro	367,--
Omikron Draw 3.0	119,--
Convector II	298,--
Phoenix	348,--
ComBase	319,--
That's Adress 2	186,--
FibuMan e	**378,--**
K-Spread 4.0	228,--
ACS	164,--
Pure Pascal EWS	318,--
Ergo	139,--
Omikron Basic 4.0 TT	a.A.
Spice v 2G5	**25,--**

BCP

Ihr Auftrag wird von uns noch am selben Tag bearbeitet und mit UPS oder per Post versandt. Beachten Sie bitte, daß nicht alle hier aufgeführten Produkte jederzeit ab Lager leiferbar sein können. "Aber wir arbeiten dran"
Die Lieferung erfolgt gegen Vorkasse oder per Nachnahme. Preisänderungen, Irrtümer und Verfügbarkeit müssen wir uns leider vorbehalten.

Drucker

NEC P 20	698,--
NEC P 60	1198,--
NEC P 70	1498,--
STAR XB 24-10	798,--
STAR ZA-200	898,--
Epson LX-400	398,--
Epson LQ-100	598,--
Epson LQ-570	748,--
Epson LQ-1070	1098,--
Epson SQ-870	1498,--
Epson SQ-1170	1758,--
Epson EPL-4100	1898,--
Epson EPL-8100	**3528,--**
HP-Laserjet II - III	a.A.
HP-Deskjet	a.A.
Canon Bubble Jet 10 ex	679,--
Canon Bubble Jet 300	**958,--**
Canon Bubble Jet 330	1328,--

1 Mbyte Simm
für alle ATARI STE's
60,--

Scanner

Epson GT 6000	**2298,--**
Interface + Software (nur in Vb.)	150,--
Epson GT 8000	3898,--
Professional Scanner II inkl. OCR	1738,--
Handyscanner 400dpi	**309,--**
Syntex v 1.2 OCR	299,--

Monitore

NEC Multisync 4FG	1678,--
NEU ProScreen TT / STE 19'	**a.A.**
Eizo 6500 21' S/W	2848,--
Eizo F550i 17' Farbe	2398,--
14' Farbe für ST	588,--
14' VGA Farbe für TT SSI/MPR	**658,--**

Hardware

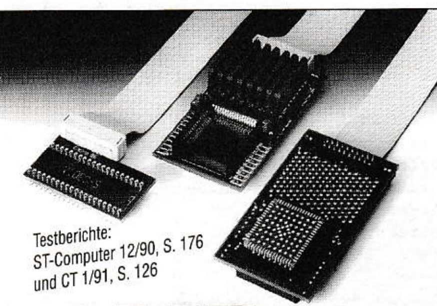
ATARI STE	a.A.
ATARI TT 030'	a.A.
Festplatte anschlußf. 105 Mbyte	1175,--
Festplatte anschlußf. 120 Mbyte	1298,--
Festplatte anschlußf. 240 Mbyte	1875,--
Wechselplatte 44 Mbyte	1248,--
Wechselplatte 88 Mbyte	1468,--
Medium 44 Mbyte	147,--
Medium 88 Mbyte	**229,--**
Streamer anschlußf. 60 Mbyte	1398,--
Floppy 1,44 Mbyte inkl. HD-Modul	219,--
Floppy 720 Kbyte	**179,--**
Floppy 5,25" 360 KB - 1,2 MB	209,--
SILENCER II Lüfterregelung	35,--

SCSI-Festplatte 48MB
komplett anschlußfertig
698,--

Diverses

ATonce 386SX/16 MHz	582,--
HBS 240 16 MHz u. 16 KB Cache	259,--
AT-Tastatur für ST - TT	264,--
That's a Mouse	59,--
Autoswitch Overscan	109,--
Crazy Dots ST/STE/TT	888,--
Chilli Digital	2318,--
TOS 2.06 für alle ST's	176,--
Speichererweiterung ab	248,--
TT-FAST RAM Karte 32/4 MB	838,--
TT-FAST RAM 32/16 MB	**1698,--**
SCSI-Controller m. Zubehör ab	168,--
Quantum 120 MB	789,--
Quantum 240 MB	1418,--
Syquest SQ 5110	798,--

BCP - Hard & Soft * Im Dorfe 19 * W-2121 Oerzen * Tel.: 04134/8689 * Fax: 8536 * BTX: BCP#



4 MByte

Speichererweiterung

Die Speichererweiertung CP 2/4 kann für alle Atar ST-Typen mit gesockelter MMU eingesetzt werden. Der Einbau ist sehr einfach: Ohne Löten oder Trennen von Leiterbahnen wird die Erweiterung in den MMU-Sockel gesteckt.

- Sehr kleine Platine (45 x 75 mm)
- 4-MBit Speicher-Technologie
- 4-Lagen Multilayer
- 100%ige Kontaktsicherheit durch Spezial-Adapter
- Speicherbausteine gesockelt

CP 2/4 ohne RAM	**Stück DM 168,–**
CP 2/4 mit 2 MByte	**Stück DM 298,–**
CP 2/4 mit 4 MByte	**Stück DM 428,–**

Bei Bestellung bitte Computertyp angeben. Eine ausführliche deutsche Einbauanleitung und ein Chip-Puller werden mitgeliefert.

CP
CP-Computer Peripherie GmbH
Entwicklung und Vertrieb
elektronischer Geräte
Viskbekstraße 14, W-4573 Löningen
Tel. 0 54 32/3 08 08, Fax 0 54 32/ 3 08 38

Versandkosten: DM 5,– bei Vorkasse, DM 10,– bei Nachnahme, Ausland: DM 15,–
Händleranfragen erwünscht

WOBO-Service
Willi B. Werk

Papyrus → →	**248,--**
Signum! 3 →	**428,--**
" + Trenndatei	469,--
" + Font Univers	513,--
" (+ Trenndatei + Font Univers)	555,--
TYPES-Trenndatei	**45,--**
Das Signum! 3 Buch	59,--
Signum! 3 Fonts	a.A.
Signum! 2	338,--
Script II (2.2)	238,--
STAD 1.3+	135,--
Piccolo	86,--
Scarabus 3	**99,--**
alle APi-Soft Prod. lieferb.	
QUERDRUCK2 →	**71,--**
That's Write 2.x	299,--
CyPress	245,--
Timeworks Publ. 2	**a.A.**
CALAMUS 1.09N	**298,--**
CALAMUS SL	1248,--
Type Art →	545,--
GFA-BASIC 3.5	216,--
GFA-BASIC 3.6 TT	259,--
ergo! →	**122,--**
GFA-ASSEMBLER	119,--
ST Pascal Plus	199,--
MAXON PASCAL	198,--
Pure C →	**318,--**
Pure Pascal *neu*	**318,--**
ACS →	**169,--**
K-SPREAD 4	219,--
K-SPREAD *light*	**88,--**
TEMPUS V.2.xx	99,--
NVDI 2	**74,--**
NVDI 2 + Kobold	**143,--**
XBoot	68,--
1st Lock →	**152,--**
1ST Base	208,--
ComBase →	**322,--**
EasyBase *A. lohnt!*	a.A.
Phoenix 2.x →	**348,--**
1st Card	278,--
1ST fibuMan	136,--
fibuMAN e/f 338,--/609,--	
ARGON 89,--/CD-V.	**a.A.**
CRYPTON	83,--
Diskus V. 2.5	136,--
MultiGEM 2	136,--
MagIX *neu* →	**119,--**
EASE; MultiDesk	je **83,--**
Harlekin II	136,--
CoCom	**126,--**
Mortimer/Mort. plus	a.A.
QFax/Pro / CodeK.	je **89,--**
Kobold, F-Copy Pro	je **75,--**
toXis → →	**59,--**
BigScreen 2 + SPEX	**83,--**
MegaPaint II prof.	249,--
Arabesque Pro	298,--
DATA light →	**98,--**
DATA DIET →	**116,--**

ZyXEL U-1496E incl.	**895,--**
Q-FAX pro (ohne Postzul.)	
ReproSt.j.+Scanman	**500,--**
TOS 2.06 Extension	**139,--**
TOS 2.06 Expansion	**148,--**
TOS-Bridge 2.06	**139,--**
Pixel Wonder	135,--
Channel Videodat De.	369,--
Hardwareprodukte von Hard&Soft; FSE; TKR; protar; vortex; etc.	**a.A.**
Speichererweit. 4MB	**369,--**
A.-Switch OverScan	110,--
That's a Mouse →	**59,--**
Das Atari 1x1 (Buch)	**49,--**
Emulatoren: eine A. lohnt!!!	

Unser PD-Angebot:
Wir bieten Ihnen die PD-Disk. aus dem Atari (PD) Journal **(J)**, PD-Pool (2000/ 5000) **(P)**, ST- Computer **(S)**, ST-Vision **(V)**, die TT- Serie **(T)**, die konTRAST-Serie **(K)** und die Demo-Serie **(De)** an. Die Preise (pro Diskette):

1 - 4 DM 5,--
5 - 9 DM 4,50
ab 9 DM 4,--

Bitte beachten Sie, daß wir nur original Fuji MF2DD Disketten (keine Bulkware) verwenden. Das alle Kopien nur mit "VERIFY" durchgeführt werden und die PD's auf Viren überprüft sind, ist für uns selbstverständlich.
Ab PD-Pool Disk. 2331 noch höhere Programmqualität; neuer Preis für PD-Pool (P) Disk. (ab 2331) DM 10,-- pro Diskette. Auch für PD-Pool-Disk. (ab 2331) wird weiterhin hochwertiges Disketten-Material verwendet.
Reine PD-Bestellungen werden bei Vorkasse versandkostenfrei und bei Zahlung per Nachnahme gegen DM 6,-- Nachnahmegebühren verschickt. Für alle anderen Bestellungen gelten die unten aufgeführten Bedingungen.
Zu PD-Versandbedingungen die fünf folgenden Artikel:

TeX 13 Disketten	**48,--**
ClipArt (16 Pool Disk.)	**56,--**
OXYD Buch + Disk.	**50,--**
OXYD2 Buch + Disk.	**60,--**
Spacola Buch + Disk.	**55,--**

Preise in DM; vorbehaltlich Irrtümer und Preisänderungen. Bei Vorkasse 2% Skonto, zuzügl. DM 5,50 Versandkostenanteil; bei Nachnahme kein Skonto, zuzügl. DM 9,50 Versandkostenanteil. Kein Ladenverkauf!
Selbstabholung nach tel. Absprache möglich!
Dies ist nur ein kleiner Ausschnitt aus unserem Angebot.

Sielwall 87, D-2800 Bremen 1
Tel. 0421/75116; Fax 0421/701285; BTX 042175116

trifolium

35 Kassel • Wilhelmsstr. 5 • TEL 0561/773077 • FAX 27963

trifolium music series

Rhythm Crack Drum Composer für den reinen Groove!	**199.–**
analyse one real time analyzing + sequence analyzing	**5998.–**

trifolium ADEQ-series

ADEQ-CAD Das universelle objektorientierte CAD-Programm	**798.–**
IEEE-488-controller	**898.–**
12 Bit Digital Transmitter Fernmessung und Digitalisierung analoger Signale	**ab 498.–**
Rainscope Datenlogger für Niederschlagsgeber	**349.–**
Wetterfax Informationssystem für Meteorologen	**598.–**

trifolium utility-series

HD-Modul	**69.–**
HD-Rohlaufwerk	**143.–**
Mailbox-System	**448.–**
Speichererw. für ATARI ST	**ab 149.–**
TOS 2.06 "switch it" für alle STs	**198.–**

SERVICE-CENTER
ATARI SYSTEM-CENTER

35 Kassel • Grassweg 14 • TEL 0561/282824 • FAX 27963

Programmierpraxis

Block-orientiertes Bildpacken

Die meisten Packformate für S/W-Bilder packen aufeinanderfolgende Bytes oder Byte-Folgen. Dies geht schnell und ist ist recht einfach, jedoch nicht für alle Fälle optimal. Das hier vorgestellte Verfahren packt nicht Byte-Folgen, sondern rechteckige Byte-Blöcke und komprimiert so einen 32-KByte-Bildschirm.

Lars Schwabe

Auf dem ST gibt es inzwischen eine ganze Anzahl verschiedener Bildformate, die teilweise mehr oder weniger gut gepackt sind (wenn überhaupt). Das hier vorgestellte Packverfahren für S/W-Bilder ist ein blockorientiertes. Es verpackt 32000-Byte-Bilder, indem es rechteckige Byte-Blöcke der gleichen Farbe (schwarz oder weiß) oder eines einfachen Musters erkennt und entsprechend codiert. Im Vergleich zum DEGAS-Packverfahren, das nur aufeinanderfolgende Bytes erkennt und codiert, ergeben sich hier bei Bildern, die nicht zu komplex sind, eindeutige Vorteile. Bei komplexeren Bildern jedoch, die nicht sehr viele Flächen gleicher Bytes beinhalten, hat DEGAS die Nase wieder vorn (wenn auch nicht sehr weit...).

Der Byte-Block

Die verwendete Bildschirmdarstellung beträgt, wie jeder weiß, 640x400 Pixel, d.h. 80x400 Bytes. Der Algorithmus soll nun gleiche Bytes zu Blöcken zusammenpacken. Deshalb bewegen sich die Werte der x-Koordinaten und der Breiten der Blöcke im Bereich von 0..79, die der y-Koordinaten und Höhen von 0..399. Leider ist dies zu viel, um in einem Byte codiert zu werden.

Um dieses Manko zu beheben, wird nun einfach das oberste Bit der x-Koordinaten, bzw. der Breiten als 9.Bit der y-Koordinaten bzw. der Höhen interpretiert, denn die x-Werte finden in 7 Bit Platz (7 Bit ergeben Werte von 0..127).

Somit wäre die Position und Größe eines Byte-Blockes auf dem Bildschirm eindeutig in 4 Bytes beschrieben. Es fehlt nur noch das Byte selbst, aus dem der Block besteht, und er ist in 5 Bytes codiert.

```
*   Blockorientiertes Packen von S/W-Bildern    *
* -------------------------------------------- *
*                  ' pack:'                     *
* EINGABE: A0 - Adresse des 32000 Bytes Screens *
*          A1 - Adresse des 'Packspeichers'     *
*                                               *
* AUSGABE: A0   - Adresse des 32000Byte Screens *
*          A1   - Adresse des 'Packspeichers'   *
*          D0.L - Länge der gepacken Datei      *
* -------------------------------------------- *
*   by Lars Schwabe   (c) 1992 MAXON Computer   *
*      (erstellt mit dem Devpac Assembler)      *
* ********************************************* *


        clr  -(sp)              * File mit den Bild-
        pea  fname              * daten öffnen
        move #$3d,-(sp)
        trap #1
        addq.l #8,sp
        move d0,fhandle
        pea screen_buffer   * Bitmap einlesen
        move.l #32000,-(sp)
        move fhandle,-(sp)
        move #$3f,-(sp)
        trap #1
        add.l #12,sp
        move fhandle,-(sp)  * File schließen
        move #$3e,-(sp)
        trap #1
        addq.l #4,sp

        lea screen_buffer,a0
        lea pack_speicher,a1
        jsr pack
        move.l a1,adresse
        move.l d0,laenge

        clr -(sp)           * File für's gepackte
        pea save_name       * Bild erstellen
        move #$3c,-(sp)
        trap #1
        addq.l #8,sp
        move d0,fhandle
        move.l adresse,-(sp)  * Daten schreiben
        move.l laenge,-(sp)
        move fhandle,-(sp)
        move #$40,-(sp)
        trap #1
        add.l #12,sp
        move fhandle,-(sp)    * File schließen
        move #$3e,-(sp)
        trap #1
        addq.l #4,sp
        clr -(sp)
        trap #1


* ------------------- *
* P A C K R O U T I N E *
* ------------------- *
pack:   movem.l d1-d7/a0-a6,-(sp)
        move.l a1,a4
        clr (a1)            * Blockzähler auf null
        addq.l #2,a4        * A4=Anfang der Daten
        lea boolean,a2
        move.l a0,a6        * Daten sichern...
        moveq #0,d7         * D6/D7 = X/Y-Zähler
pl1:    moveq #0,d6         * (aufwärts zählend)
pl2:    move.l a6,a3
        move.b (a6)+,d0     * Byte heraussuchen...
        cmp.b #1,(a2)       * wurde das Byte schon
        beq pl_end          * bearbeitet???

* größtes Rechteck ab A3 nach D4-D5 schaffen
        move.b d0,byte
        move.l a2,a5
        clr flaeche
        clr hoehe
```

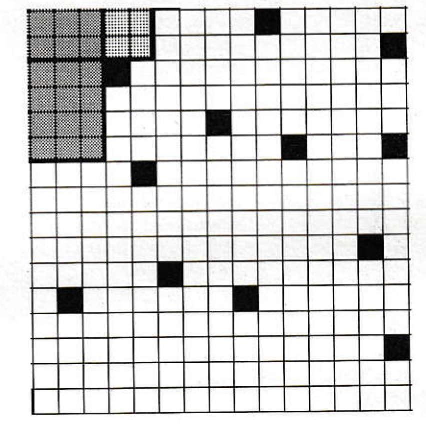
→



Auf der Suche nach dem goldenen Block ...

Nun geht es ans Erkennen der Blöcke. Hierzu wird einfach, ausgehend von jedem Byte des Bildschirms, das die linke obere Ecke des Blockes bilden soll, das Rechteck mit dem größten Flächeninhalt berechnet, es sei denn, das zu untersuchende Byte ist schon in einem Block codiert.

Dieser Suchvorgang ist leider nicht so einfach, wie es scheint. Versuchen Sie sich doch einmal einen Algorithmus zu überlegen, der, ausgehend von einem weißen Karo, in einem Kreuzworträtsel das Rechteck mit dem größten Flächeninhalt findet. Die schwarzen Fragefelder wären hierbei die Begrenzung.



Rechtecke ab der linken oberen Ecke. Das graue hat die größte Fläche.

Mein Algorithmus geht folgendermaßen vor: Er berechnet die Breite eines möglichen Blockes, ausgehend von der linken oberen Ecke (das momentan zu untersuchende Byte). Mit dieser Breite und der Höhe des Blockes (am Anfang natürlich nur 1) wird der Flächeninhalt dieses Rechteckes berechnet. Dann wird die Breite des Blockes eine Zeile weiter unten gemessen. Mit diesem Wert, der max. die Breite der zuvor untersuchten Zeile annehmen kann, wird erneut der Flächeninhalt berechnet.

Ist dieser Flächeninhalt größer als der bis dahin größte Flächeninhalt, wird er als akt. Wert für diesen Vergleich gesichert. Auch die Breite und Höhe dieses Blockes werden als momentan beste Werte gesichert.

Der Anfangswert der Fläche A ist natürlich für den ersten Durchgang extra klein gewählt.

So wird mit jeder Zeile verfahren, bis die Breite Null wird, sich also ein anderes Byte unter dem Block befindet, an die Grenzen des Bildschirms oder eines anderen Blockes gestoßen wird.

Sind alle Blöcke auf diese Weise gefunden, werden die restlichen Bilddaten einfach linear hinter diese Blockliste geschrieben. Da alle Bytes, die zu Blöcken zusammengefaßt sind, in einem Boolschen Feld markiert sind, fällt es nun leicht, die unverpackten Bilddaten ausfindig zu machen.

Das Entpacken

Vor der Blockliste steht in einem Wort (16 Bit) die Anzahl der Blöcke dieses Bildes. Dann können beim Entpacken der Bilder alle Blöcke in den Bildschirmspeicher kopiert und in einem Boolschen Array markiert werden. Dies ist wichtig, um nun mit den anderen Bilddaten die entstandenen Lücken zu füllen. Zu beachten ist jedoch, daß das oberste Bit der x-Werte als das 9.Bit der äquivalenten y-Werte zu interpretieren ist.

Vielleicht experimentieren Sie ein wenig mit dieser Routine und erweitern sie um die Fähigkeit, Blöcke aus zwei oder mehr Bytes zu erkennen ...

```
          moveq #81,d1 * D1 = Spalten bis zum
          sub d6,d1    *      rechten Rand
          move #401,d2 * D2 = Zeilen bis zum
          sub d7,d2    *      unteren Rand
          moveq #-1,d4 * D4 = vorige Breite
          moveq #1,d5  * D5 = Höhe (anfangs 1)
          moveq #1,d3  * D3 = akt. Breite
bl1:      cmp d1,d3
          beq failed   * rechten Rand erreicht?
          cmp d2,d5
          beq ende     * unterer Rand erreicht?
          cmp.b (a3,d3),d0
          bne failed
          cmp.b #1,(a5,d3) * ist das Byte schon
          beq failed       * in einem Block ???
          addq #1,d3 * rechts war noch ein Byte
          bra.s bl1
failed:   cmp #0,d3    * ist der Block nach unten
          beq ende     * hin abgeschlossen ?
          cmp d3,d4
          bhi neue_b
          move d4,d3       * neue Breite > alte
neue_b:   move d3,d4
          mulu d5,d3       * (Fläche) A = D5 * D3
          cmp flaeche,d3   * ist diese kleiner
          bls f_kleiner    * als die größte ?
          move d3,flaeche
          move d5,hoehe    * (bessere Höhe &  A)
```

```
f_kleiner: lea 80(a3),a3   * neue Zeile
           lea 80(a5),a5   * neue 'Boolean-Reihe'
           addq #1,d5
           moveq #0,d3  * Breite der akt. Zeile 0
           bra.s bl1

* nun muß die passende Breite ermittelt werden
ende:      moveq #0,d4
           move flaeche,d4
           move hoehe,d5
           divu d5,d4

* lohnt es sich überhaupt den Block einzutragen?
* d.h. (Fläche) A > 5
           move d4,d0
           mulu d5,d0
           cmp #5,d0
           bls pl_end
           addq #1,(a1)     * Blockzähler erhöhen
           move.b d6,(a4)+ * X & Y eintragen
           move.b d7,(a4)+
           btst #8,d7
           beq ybit8
           bset #7,-2(a4)   * das 7 Bit setzen
ybit8:     move.b d4,(a4)+ * Breite/Höhe schreib.
           move.b d5,(a4)+
           move.b byte,(a4)+ * Byte eintragen
```

→




```
                btst #8,d5
                beq fill          * 7. Bit der Breite
                bset #7,-3(a4)    * als 8. Bit der Höhe

* im 'Boolean-Feld' die Bytes des Blocks markieren
fill:           subq #1,d4        * Breite
                subq #1,d5        * Höhe
                move.l a2,a5      * A5 = Pointer auf den
f_loop2:        move d4,d0        *      ersten Wert
f_loop1:        move.b #1,(a5,d0)
                dbf d0,f_loop1
                lea 80(a5),a5
                dbf d5,f_loop2

pl_end:         addq.l #1,a2    * Boolean-Pointer erhöhen
                addq #1,d6      * X-Schleife schließen
                cmp #80,d6
                bne pl2
                addq #1,d7      * Y-Schleife schließen
                cmp #400,d7
                bne pl1

* alle Blöcke sind eingetragen, jetzt werden
* die restlichen Daten gesucht...
                move (a1),d5
                mulu #5,d5
                addq.l #2,d5      * in D5 wird die
                move.l a0,a3      * Länge berechnet
                lea boolean,a5
                move #399,d1
de_loop2:       moveq #79,d0
de_loop1:       cmp.b #1,(a5,d0)
                beq de_check1
                addq.l #1,d5  * Dateilänge erhöhen
                move.b (a3,d0),(a4)+
de_check1:      dbf d0,de_loop1
                lea 80(a3),a3     * neue Bildschirmzeile
                lea 80(a5),a5     * neue 'Boolean-Zeile'
                dbf d1,de_loop2
                move.l d5,d0      * Länge der Datei
                movem.l (sp)+,d1-d7/a0-a6
                rts


* ************ *
* DATA-Segment *
* ************ *
            data
fname       dc.b "a:\bilder\olympic.art",0
*                Filename der 32000er-Bitmap

            even
save_name dc.b "a:\bilder\olympic.com",0
*                gepacktes Bild

            even

* *********** *
* BSS-Segment *
* *********** *
            bss
* vom Hauptprogramm
ssp         ds.l 1
adresse     ds.l 1
laenge      ds.l 1
fhandle     ds.w 1
screen_buffer ds.l 8000       * geladenes Bild
pack_speicher ds.l 8000       * gepackte Bilddaten

* ab hier zur Packroutine
boolean     ds.l 8000     * Boolschen Array zum Packen
hoehe       ds.w 1
flaeche     ds.w 1        * Flächeninhalt eines Blocks
byte        ds.b 1        * Byte des Blocks
```

```
*                     ENTPACKROUTINE                     *
* ------------------------------------------------------ *
*   by Lars Schwabe  (c) 1992 MAXON Computer             *
* ****************************************************** *

flaenge  = 10206    * hier die Filelänge eintragen

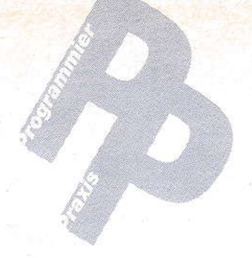
* gepackte Bilddaten lesen
            clr  -(sp)            * Datenfile öffnen
            pea  fname2
            move #$3d,-(sp)
            trap #1
            addq.l #8,sp
            move d0,fhandle
            pea pack_speicher    * Bitmap einlesen
            move.l #flaenge,-(sp)
            move fhandle,-(sp)
            move #$3f,-(sp)
            trap #1
            add.l #12,sp
            move fhandle,-(sp)   * File schließen
            move #$3e,-(sp)
            trap #1
            addq.l #4,sp

            dc.w $a00a
            move #$03,-(sp)
            trap #14
            addq.l #2,sp              * log. Screen holen
            move.l d0,a0              * ZIEL-Adresse
            lea pack_speicher,a1 * QUELL-Adresse
            jsr repack
            dc.w $a009

            move #$07,-(sp)          * WARTEN...
            trap #1
            addq.l #2,sp
            clr -(sp)
            trap #1


* -------------------------- *
*E N T P A C K R O U T I N E *
* -------------------------- *
repack:     movem.l d0-d1/d3-d4/a0-a4,-(sp)
            lea boolean,a3     * Basis des Arrays
            move (a1)+,d3      * D3 = Anz. der Blöcke
            subq #1,d3
loop1:      moveq #0,d6
            moveq #0,d7
            move.b (a1)+,d6  * X-Koordinate
            move.b (a1)+,d7  * Y-Koordinate
            btst #7,d6
            beq check1
            bset #8,d7    * 8.Bit bei Y setzen
            bclr #7,d6    * 7.Bit bei X löschen
check1:     moveq #0,d4
            moveq #0,d5
            move.b (a1)+,d4     * Breite
            move.b (a1)+,d5     * Höhe
            btst #7,d4
            beq check2
            bset #8,d5    * 8.Bit der Höhe 1
            bclr #7,d4    * 7.Bit der Breite 0
check2:     move.l a0,a2 * (Entpackspeichers)
            move.l a3,a4 * (Boolschen Felde)
            mulu #80,d7
            add d7,a2
            add d6,a2    * A2 = Startbyte im
            add d7,a4    *      32000er-Screen
            add d6,a4    * A4 = gleiches Byte, jedoch
            subq #1,d5   *      im boolschen Feld
            subq #1,d4
            move.b (a1)+,d0
            movem d4-d5,-(sp)  * D4/D5 sichern
loop3:      move d4,d1
loop2:      move.b d0,(a2,d1)  * Byte des Blocks
            dbf d1,loop2       * eintragen...
            lea 80(a2),a2
```

→




```
          dbf d5,loop3

* den Block im boolschen Array markieren,
* um die restlichen Bytes zu finden
          movem (sp)+,d4-d5
f_loop2:  move d4,d0
f_loop1:  move.b #1,(a4,d0)
          dbf d0,f_loop1
          lea 80(a4),a4
          dbf d5,f_loop2
          dbf d3,loop1     * ab zum nächsten Block

* die Blöcke sind eingetragen, jetzt noch die
* Grafikdaten linear in die Lücken schreiben
            move #399,d5
grafik_2:   moveq #79,d0
grafik_1:   cmp.b #1,(a3,d0)
            beq gcheck
            move.b (a1)+,(a0,d0)
gcheck:     dbf d0,grafik_1
            lea 80(a3),a3
            lea 80(a0),a0
            dbf d5,grafik_2
            movem.l (sp)+,d0-d1/d3-d4/a0-a4
                 rts


* ************ *
* DATA-Segment *
* ************ *
                 data
fname2           dc.b "a:\bilder\starp.com",0
*                         Filename des gepackten Files

                 even

* *********** *
* BSS-Segment *
* *********** *
                bss
ssp              ds.l 1
fhandle          ds.w 1
screen_buffer ds.l 8000   * geladenes Bild
pack_speicher ds.l 8000   * gepackte Bilddaten

boolean          ds.l 8000
* (um die noch nicht bearbeiteten Bytes zu finden,
*  denn die Blöcke sind hier markiert)
```

Programmierpraxis

AT
Watchtower im Atari

In der UNIX-Welt ist die Hintergrundverarbeitung von Kommandos gang und gebe. Einen wichtigen Bestandteil zur Büroautomatisation bietet der Befehl AT. Auf ihn bauen Mail-Systeme, Backup-Lösungen und andere komplexe Büroaufgaben auf. Auch auf dem Atari sollte es möglich sein, für die vielen vorhandenen Shells ein solches Tools aufzubauen. Natürlich sind dabei einige Klippen des TOS zu umschiffen.

Harald Wörndl-Aichriedler

Der Befehl AT wird meistens in ähnlicher Syntax aufgerufen, wie man einen kurzen Satz formulieren würde, z.B. „at 13:44 print test.c“. In dieser Implementierung wird vor dem auszuführenden Befehl noch als eindeutige Trennung zu den Optionen ein großgeschriebenes DO eingefügt. Es soll verhindern, daß falsch geschriebene Optionen oder Zeiten als Befehl angesehen werden. Die gesamte dokumentierte Syntax steht in den Anfangskommentaren des Listings.

AT soll nun im Hintergrund warten und einen Befehl aus einer Liste ausführen, falls er zum jetzigen Zeitpunkt auszuführen ist. Dabei wird das Standard-C-Kommando *System* verwendet. Hier kommt auch schon die erste Einschränkung. AT benötigt eine Shell, die über die Systemvariable *_shell_p* das Kommando System zugänglich macht. Da der Aufruf auch in GEM-Programmen funktioniert, ist es ratsam, keine Kommandos aufrufen zu lassen, die eine Bildschirmausgabe machen, sondern entweder darauf zu verzichten, oder sie umleiten zu lassen („>“). Eine Alternative dazu ist Gemini mit der Mupfel-Shell im GEM-Fenster. Hier kann man, solange man sich in Gemini befindet, auch Bildschirmausgaben über *stdout* machen lassen, die dann im Mupfel-Fenster erscheinen. Dazu wird vom Wecker das auszuführende Programm über das Gemini-Protokoll, und nicht über *_shell_p* aufgerufen. Es ist ohne Gemini auch nicht ratsam, Programme (nicht interne Befehle) über die Shell aufzurufen, da dann Bomben am Bildschirm nicht auszuschließen sind. Man sollte seine eigene Shell darauf testen, um Fehler zu vermeiden.

Ein Problem stellt die Funktion zum Testen des Vorhandenseins von Gemini dar, da es in allen aufgerufenen Program-

```
/*
    AT.C

    (c) 1992 MAXON Computer

    Stellt Wecker auf eine gewisse Zeit.
    Syntax:
    at [-v] [-n] [hh:mm [dd.mm.jjjj]]
        [+###{s[econd] | mi[nute] | h[our]
        | d[ay] | mo[nth] | y[ear]}] DO
        'kommandozeile'

    -v: Verbose, volle Information (nur falls -n,
         sonst immer)
    -n: Keine Sicherheitsabfrage

*/

#include <tos.h>
#include <ext.h>
#include <time.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <aes.h>

char usage[] = "\x1bp Anwendung: \x1bq\n"
"at [-v] [-n] [hh:mm [dd.mm.jjjj]] [+###{s[econd]
  | mi[nute] | h[our]\n"
"   | d[ay] | mo[nth] | y[ear]}] DO
  'kommandozeile'\n"
"-Reihenfolge beachten!\n";


int appl_id = -1;

#define TRUE   -1
#define  FALSE 0
int verbose = FALSE, no_req = FALSE;

#define WORK_DIR  "C:\\WECKER\\"
#define KOMM_STR  WORK_DIR "TIME%04i.MUP"

#define WECK_MSG  2422  /* Nachrichtennummer für
                           Weckerstart */
#define WECK_REQ  2423  /* Nachrichtennummer für
                           Rückgabewert */

typedef struct
{
   time_t   next, offset;
   int kommando;
} wecke;

void end_error(char *str);
int msg_num(char *msg);
void del_batch_file(int kommando_nr);
int request(void);
void send_msg(wecke *msg);

main(int argc, char *argv[])
{
   time_t   exec_time;
   struct tm zeit = {0,0,0,0,0,0,0,0,0};
   int comm_beg, search = 0, i;
   long offset=0L;
   char line[128]="\0";
   wecke evnt;

   /*** SUCHE BEGIN DER AUFZURUFENDEN
        KOMMANDOZEILE: ***/
   for(comm_beg=0; comm_beg<argc &&
               strcmp(argv[comm_beg], "DO");
               comm_beg++);
   comm_beg++;
   if(comm_beg >= argc || comm_beg <2)
      end_error(usage);

   /*** WERTE ARGUMENTE AUS: ***/
   while(*argv[++search] == '-')
      switch(argv[search][1])
      {
         case 'N':   case 'n':   no_req = TRUE;
         break;
         case 'V':   case 'v':   verbose = TRUE;
         break;
         default: printf("Falsche Option: %s\n",
         argv[search]);
```

→



men fehlerhafterweise über *appl_find* zu finden ist. Deshalb wird über die aktuelle Basepage getestet, ob Gemini aktiv ist. Andere Shell-Besitzer können diese Teile auslassen.

Um AT von der Kommandozeile und aus Stapeldateien zu bedienen, ist es in zwei Teile getrennt, was außerdem Speicherplatz spart. Der Steuerungsteil ist eine normales TTP-Programm, das die Auswertung der Zeiteingabe übernimmt. Sinnigerweise ist dieses Tool mit „AT" benannt. Die Zeitüberwachung und Ausführung, der zweite, nicht weniger wichtige Teil des Utilities, ist das Accessory „WECKER.ACC", es überwacht die Liste mit den Zeitereignissen. Um sich die Zusammenarbeit der beiden Teile besser vorstellen zu können, will ich den Ablauf einer Zeiteintragung nun anhand eines Beispiels veranschaulichen: Zuerst wird vom Benutzer AT.TTP aufgerufen. Er gibt dabei zum Beispiel „at 16:55 DO copy c:*.c d:" ein. In AT wird nun die Zeit in Sekunden seit dem 1.1.1970 GMT berechnet, und nach einer eventuellen Sicherheitsabfrage das Kommando in eine numerierte Batch-Datei in ein Wecker-Verzeichnis kopiert. Danach sendet AT eine AES-Message an WECKER und wartet mit einem Timeout auf die Empfangsbestätigung von WECKER. Tritt ein Fehler auf, wird dieser ausgegeben, und die Batch-Datei mit dem Kommando gelöscht.

WECKER empfängt eine Nachricht von AT und versucht, das Zeitereignis in seiner internen statischen Liste einzuordnen. Gelingt das, wird die Eintragsnummer zurückgegeben, ansonsten der Fehlercode. Das Eintragen des Ereignisses ist somit abgeschlossen. Nun läuft WECKER normal im Hintergrund weiter und überprüft alle 1,5 Sekunden, ob ein Ereignis in der Liste auszuführen ist. Wenn ja, wird kontrolliert, ob überhaupt eine Shell vorhanden ist. Erst dann kann das Kommando ausgeführt werden. Dazu wird einfach ein String mit dem Namen der numerierten Batch-Datei an die Shell übergeben, die diese dann ausführt.

Beim vorigen Beispiel würde die Shell also alle C-Sources von c:\ nach d: kopieren. Praktisch für ein Backup der Daten. Es hat jedoch wenig mit Automatisation zu tun, wenn man jedes auszuführende Kommando einmal mit AT aktivieren müßte. Deshalb habe ich noch die *every*-Option in die Programme integriert. Gibt man nach der Zeit bzw. nach dem Datum noch die Option *+1day* ein, wird dieses Backup ab sofort täglich gemacht. Als Einheit hinter der Zahl kann entweder second, minute, hour, month oder year stehen. Es werden die Zahlenwerte dann dementsprechend auf Sekundenwerte mit der Einheit hochmultipliziert. Dabei ist die Dauer eines Jahres folgendermaßen definiert: alle 4 Jahre ein Schaltjahr mit 366 Tagen, das alle 16 Jahre ausfällt. Ein Monat entspricht 1/12 Jahr. Deshalb sollte es nicht verwundern, daß eine langfristige Zeitdifferenz fast immer um ein paar Stunden in die Hose geht, (und das nach Murphy's Gesetz natürlich besonders bei wichtigen Ereignissen!).

In den beiden Programmen sind keine besonderen Techniken vorhanden. Erwähnenswert ist vielleicht der Nachrichtenaustausch über das AES, der sicher sehr zum Nachbau in eigenen Applikationen zu empfehlen ist, da er weder XBRA, XREF, XARG, ARGV oder COOKIES benötigt, wobei jedoch eine offizielle Registrierung der belegten Nachrichtennummern sicher von Nutzen wäre, um Kollisionen zu verhindern.

Auch die Speicherverwaltung stellt in Accessories eine Hürde dar, da ein einfaches Malloc zu schweren Problemen

```
                        end_error(usage);
        }
    if(!no_req) verbose = TRUE;

    /*** FALLS ZUMINDEST ZEIT GEGEBEN: ***/
    if(strchr(argv[search],':'))
    {
        sscanf(argv[search++],"%2i:%2i",
        &zeit.tm_hour,&zeit.tm_min);

        /*** FALLS AUCH DATUM DABEI: ***/
        if(strchr(argv[search],'.'))
        {
            sscanf(argv[search++],"%2i.%2i.%4i"
                ,&zeit.tm_mday, &zeit.tm_mon,
                &zeit.tm_year);
            zeit.tm_mon--;
            if(zeit.tm_year > 1900) zeit.tm_year -=
                                                1900;
        }
        else
        {
            /*** SONST DATUM VON HEUTE: ***/
            struct date act_date;
            getdate(&act_date);
            zeit.tm_mday = act_date.da_day;
            zeit.tm_mon = act_date.da_mon-1;
            zeit.tm_year = act_date.da_year-1900;
        }

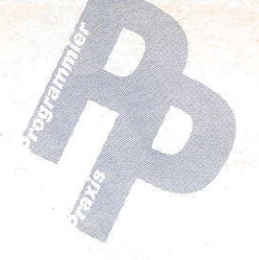
        /*** ZEIT DARF NICHT IN VERGANGENHEIT
             LIEGEN: */
        exec_time = mktime(&zeit);
        if(exec_time < time(NULL))
          exec_time = time(NULL);
    }
    else
      /*** FALLS GAR NICHTS GEGEBEN, VERWENDE
             AKTUELLE ZEIT ***/
      exec_time = time(NULL);

    if(verbose)                 /* INFO AUSGEBEN: */
    {
      zeit = *gmtime(&exec_time);
      printf("\x1bp AT: Eventsteuerung \x1bq\n"
         "Zeit: %i:%02i  Datum: %i.%i.%i\n"
         ,zeit.tm_hour+1, zeit.tm_min
         ,zeit.tm_mday, zeit.tm_mon+1,
          zeit.tm_year+1900);
    }

    /*** OFFSET BERECHNEN: ***/
    if(argv[search][0] == '+')
    {
        char s[20];
        sscanf(argv[search]+1, "%li%s", &offset,s);
        strupr(s);
        switch(*s)
        {
            case 'S': break;  /* Sekunde */
            case 'M':
                switch(s[1])
                {
                    case 'I': offset *= 60L; break;
                    /* Minute */
                    case 'O': offset *= 2629350L;
                    break; /* Monat */
                    default: offset = 0L;
                }; break;
            case 'H':   offset *= 3600L; break;
            /* Stunde */
            case 'D':   offset *= 86400L; break;
            /* Tag */
            case 'Y':   offset *= 31552200L; break;
            /* Jahr */
            default: if(*s>=32) offset = 0L;
        }
        if(offset == 0)
        {
            printf("Falsche Zeiteingabe: %s\n",
                    argv[search]+1);
            end_error(usage);
        }
    }


```

→



führt, weil der Speicher nicht auf Dauer zur Verfügung steht. Durch die Trennung von Verwaltung und Ausführung ist man hier jedoch mit einer statischen Liste gut bedient, da das Datenaufkommen kaum mehr als 200 Ereignisse zu je 10 Bytes beträgt. Diese 2 KBytes sind bei einer heutigen zig-MB Maschine wohl kaum mehr der Rede wert.

Im allgemeinen kann man folgende Sparmaßnahmen in TC/PC-Compilaten vornehmen: Erstens sollte man den Stack möglichst klein bemessen. 1-2 KBytes reichen bei den meisten Anwendungen vollkommen. Nur bei einigen *stdio*-Kommandos kann es zu Problemen kommen, auf die man aber elegant durch einen Datenverlust oder ein paar schöne Bomben aufmerksam gemacht wird. Sehr gefährlich sind hier die Funktionen printf und Pendanten, die minimal 2 KBytes benötigen.

Eine weitere Maßnahme ist das Weglassen der FP-Library, da dann automatisch die Format-Funktionen (*printf* & Co) fast nur noch die Hälfte des Speicherbedarfes haben.

Die statische Liste mit den Ereignissen ist geordnet und besitzt Elemente der Struktur wecke. Dabei wird der Vorteil von C genutzt, einer strukturierten Variablen eine andere zuzuweisen. Eine sehr elegante Art der Programmierung, die die Programme kürzer und übersichtlicher gestaltet.

Die Programme sind im allgemeinen schon sehr abgespeckt und verlangen deshalb nach Erweiterungen. Bitte achten Sie darauf, daß Sie keine Zugriffe auf Systemvariablen und unsichere Interrupt-Programmierung einbauen, da dann die Kompatibilität zu MultiTOS und Derivaten nicht mehr gegeben ist. Arbeiten Sie stattdessen an leistungsfähigen Shellscripts, oder, falls Sie noch keine Shell besitzen, eben an einer Shell. Informationen dazu sollte die PC-Help-Funktion zu *System* oder diverse andere Lektüre bereitstellen.

```
    /*** REST VON KOMMANDOZEILE ZUSAMMENFÜGEN: ***/
    for(strcat(line,argv[i=comm_beg]); i<(argc-1);
       strcat(line,argv[++i]))
       strcat(line," ");

    if(verbose) printf("Kommandozeile: %s\n",
       line);
    if(verbose && offset)
       printf("Ausführung alle %li Sekunden.\n",
              offset);

    if((i=msg_num(line)) < 0)
       end_error("Konnte Datei in "WORK_DIR" nicht
                 öffnen!\n");

    /*** WECKSTRUKTUR AUSFÜLLEN: ***/
    evnt.next = exec_time;
    evnt.offset = offset;
    evnt.kommando = i;

    /*** SICHERHEITSABFRAGE: ***/
    if(!no_req)
    {
       printf("Sicherheitsabfrage! Falls alles
              stimmt, drücken Sie ':'");
       if(getchar() != ':')
          end_error("Sie haben die Operation
                    abgebrochen...\n");
    }

    appl_id = appl_init();

    /*** NACHRICHT SENDEN: ***/
    send_msg(&evnt);
    if(verbose) printf("Nachricht an Wecker
                       gesendet...");

    /*** BESTÄTIGUNG ERWARTEN: */
    switch(i=request())
```

```
    {
       case -1:
          del_batch_file(evnt.kommando);
          end_error("Zeitpuffer voll. Andere
                    Ereignisse Stornieren!\n");
       case -2:
          del_batch_file(evnt.kommando);
          end_error("WECKER.ACC antwortet
                    nicht!\n");
       default:
          if(verbose)
             printf("Empfang bestätigt. Noch %i
                    Ereignisse davor.\n"
                    "Alles OK\n", i);
    }
    appl_exit();
    return 0;
}

void end_error(char *str)
{
    printf("%s",str);
    if(appl_id >= 0) appl_exit();
    exit(0);
}

int msg_num(char *msg)
{
    char f_name[128];
    int t, han;

    for(t=0; t<10000; t++)
    {
       sprintf(f_name, KOMM_STR, t);
       if(Fsfirst(f_name,0))
       {
          han = Fcreate(f_name,0);
          if(han<0)
             {t=-1; break;}
          Fwrite(han, strlen(msg), msg);
          Fclose(han);
          if(verbose)
             printf("Zum löschen der Weckzeit
                    löschen Sie %s\n", f_name);
          break;
       }
    }
    return t;
}

void del_batch_file(int kommando_nr)
{
    char dateiname[128];
    sprintf(dateiname, KOMM_STR, kommando_nr);
    Fdelete(dateiname);
}

int request(void)
{
    int in[8], d, e_art;
    do
    {
       e_art = evnt_multi(MU_TIMER | MU_MESAG,
               0,0,0,0,0,0,0,0,0,0,0,0,0,
               in, 3000, 0, /* Low/High-Counter
                                Timer */
               &d,&d,&d,&d,&d,&d);

    } while(e_art != MU_TIMER &&
          !(e_art == MU_MESAG && in[0] ==
          WECK_REQ));
    if(e_art == MU_TIMER)
       return -2;
    return in[3];
}

void send_msg(wecke *msg)
{
    int ap_id = appl_find("WECKER  ");
    int buf[8] = {WECK_MSG,0,0};

    if(ap_id < 0)
       end_error("WECKER.ACC muß unter diesem
                 Namen gestartet sein!\n");
    buf[1] = appl_id;
    *(wecke*)(&buf[3]) = *msg;
    appl_write(ap_id, 16, buf);
}
```

Text-Hardcopy

Hardcopy-Problem auf dem Atari ST (ALT - HELP drücken und sich wundern) ist eigentlich durch eine große Anzahl von Hardcopy-Utilities, Accessories etc. ausreichend gelöst. Warum noch ein weiteres Programm? Alle mir bekannten Hardcopy-Programme arbeiten nach dem gleichen Prinzip: Man kann Teile des Bildschirms (bzw. den ganzen) als Grafik ausdrucken bzw. auf Diskette speichern. Diese Hardcopy-Programme unterscheiden sich lediglich in ihrem Funktionsumfang bzw. ihrer Bedienbarkeit.

Andreas Lauterbach

Ich fertige Hardcopies normalerweise nur aus dem Grund an, weil mich die Informationen in einem Fenster (Directory, Help-Screen) interessieren. Es kommt dabei nicht auf grafische Qualität der Hardcopy bzw. die Darstellung von Icons etc. an. Eine Grafik-Hardcopy ist in diesem Falle sogar eher als störend zu betrachten; denn Grafikdruck dauert länger als Textdruck, ist in der Qualität durchweg schlechter (zumindest auf normalen Druckern), kostet mehr Farbband und Nerven (Lärm!) etc. Zudem lassen sich Bildausschnitte durchweg schlechter weiterverarbeiten als Texte.

So wurde die Idee geboren, den zu druckenden Text von Grafik in eine ASCII-Datei zu wandeln, die entweder gedruckt oder in Textprogramme übernommen werden kann. Diese Aufgabe sollte eine Art „Mini-OCR“ übernehmen. Die Bezeichnung OCR ist schon fast übertrieben, schließlich sehen auf dem Bildschirm alle Buchstaben gleichen ASCII-Codes auch gleich aus. Auch die Abstände der Zeichen voneinander sind klar definiert. Lediglich die Position des Textes auf dem Bildschirm ist frei verschiebbar (Fenster!). Der Abstand der Textzeilen ist davon abhängig, ob der Text in einem Fenster steht (17 Pixel) bzw. auf einem TOS-Screen erscheint (15 Pixel). Diesem Umstand wird bei Programmstart Rechnung getragen, indem das Programm erfragt, ob man ein Fenster (GEM) oder einen Text-Screen (TOS) hardkopieren möchte.

```
' Text-Hardcopy
'(c) 1992 MAXON Computer
'
$m 33000                ! Speicher reservieren
ap_id&=APPL_INIT()
'
' Programm läuft nur in der hohen Auflösung (640x400) und mit
' dem normalen Systemzeichensatz (8x16) Pixel. Attribute, wie
' fett, kursiv etc. werden nicht erkannt...
'
' Das Programm kann auf 3 Arten gestartet werden :
' 1. in der Menüleiste anwählen
' 2. durch Drücken und Halten der beiden Shift-
    Tasten (ca. 1 sec.)
' 3. durch Umbennen in *.PRG und Starten von einer Shell
'
CLR charset$
GET 20,0,40,20,backgr$
FOR t|=0 TO 158                    ! Zeichentabelle initialisieren
  TEXT 30,13,CHR$(t|)              ! Diese Routine wird nur einmal
  GET 30,0,36,15,char$             ! beim Booten des Rechners bzw. beim
  charset$=charset$+char$          ! Start als PRG durchlaufen!
NEXT t|                            ! Die Obergrenze (ASCII 158) kann
PUT 20,0,backgr$                   ! heraufgesetzt werden. Scan-Vorgang
CLR backgr$                        ! dauert dann aber länger!
IF ap_id&<>0
  me_id&=MENU_REGISTER(ap_id&," Texthardcopy ")
  DO
    b$=SPACE$(16)                  ! Ereignispuffer
    a%=VARPTR(b$)
    '
    ' Timer event (1 sek.) setzen und auf Ereignis warten
    '
    ~EVNT_MULTI(48,0,0,0,0,0,0,0,0,0,0,0,0,0,a%,1000)
    '
    ' Menüleiste abfragen
    IF ASC(MID$(b$,2,1))=40  ! Accessory angeklickt?
      hardcpy
    ENDIF
    '
    r%=BIOS(11,-1)         ! Shift-Tasten gedrückt?
    IF (r% AND 3)=3
      hardcpy
    ENDIF
  LOOP                     ! Acc's enden nie ...
ELSE
  hardcpy                  ! oder als PRG gestartet?
  END
ENDIF
PROCEDURE hardcpy
  SHOWM
  CLR box_wide,snap!,anz_char%,txt$
  ' erst mal ein paar allgemeine Fragen stellen
  ALERT 2,"Texthardcopy by A.Lauterbach|Wovon soll eine
  Text-Hardcopy|angefertigt werden ?",1," GEM | TOS | Help ",d%
  SELECT d%
  CASE 1
    add_y%=3
  CASE 2
    add_y%=1
  CASE 3
    ALERT 1,"GEM - Hardcopy von Window|TOS - Hardcopy von TOS-Screen|Select - Zeilenabstand wählen|Snap - Snapshotutility",1,"Okay|Select|Snap",d%
    IF d%=2
      REPEAT
        ALERT 1,"Zeilenabstand wählen| |Abstand =
```

→



Funktionsweise

Da das Aussehen eines jeden Zeichens genau definiert ist, und auch keine zwei Zeichen identisch aussehen, bietet sich folgendes vorgehen an: Die Grafikmuster der einzelnen Zeichen (8x16 Pixel, entspricht 38 Byte Grafikdaten in GFA-BASIC) werden in einem String hintereinander gespeichert. Die absolute Position der Grafikdaten entspricht also dem 38-fachen des entsprechenden ASCII-Codes. Durch byteweisen Vergleich mit dem gesuchten Muster kann also das entsprechende Zeichen wiedererkannt werden. Dieser Vergleich geschieht mit der INSTR-Funktion, die die Position eines Strings in einem anderen wiedergibt. Liefert die INSTR-Funktion einen Wert >0, wurde ein bekanntes Grafikmuster gefunden, liefert sie den Wert 0, war das Grafikmuster nicht bekannt. Glücklicherweise ist die Funktion eindeutig definiert, so daß die „Erfolgsrate" der Mini-OCR bei 100% liegt (es treten keine Scan-Fehler auf).

Voraussetzung für den Scan-Vorgang ist die Kenntnis, wo der Text auf dem Bildschirm beginnt. Diese Position ist bekanntlich ja pixelweise frei verschiebbar. Der Bildschirm wird hierzu innerhalb des in Frage kommmenden Bereichs pixelweise abgetastet, bis ein bekanntes Grafikmuster (ein bekanntes Zeichen) gefunden wird. Der nächste Buchstabe steht dann mit Sicherheit 8 Pixel weiter rechts, die nächste Zeile beginnt mit Sicher- heit 15 oder 17 Pixel weiter unten.

```
D:\GFABASIC\
415788 Bytes in 18 Dateien.
  GRAPHIC                   01-03-91  12:28
  MICHEL                    14-04-91  11:24
  PLOTTER                   01-03-91  12:33
  PROJEKT                   01-03-91  12:30
  SIMULA                    01-03-91  12:34
  SOURCE                    01-03-91  12:17
  COMPILER PRG    18819     25-04-90  22:01
  DRAW     LST     1482     30-05-91  21:49
  GFA3BLIB       118918     25-04-90  22:00   ——— Gummiband
  GFA3BLIB NDX    27406     25-04-90  22:00
  GFABASIC PRG    92568     16-04-90  14:50
  GFABAS_2 PRG    57613     18-08-87  16:31
  GFA_BCOM PRG    70460     25-04-90  22:00
  GL       PRG     2901     25-04-90  22:01
  GR_PATHS DAT       14     31-07-90  21:59
  MENUX    PRG    21747     08-05-91  21:39
```

So sollte das Gummiband angelegt werden

Vorraussetzungen

In dieser Version läuft das Programm nur in der hohen Auflösung und mit dem aktuellen Systemzeichensatz (8 x 16 Pixel), d.h. der Atari Standardkonfiguration. Schriftattribute wie fett, kursiv etc. werden nicht erkannt. Diese Einschränkungen sind jedoch sicherlich zu verschmerzen, zumal sie in „normalen" Fenstern ohnehin nicht verwendet werden. Nicht erkannte Zeichen werden durch ein Space ersetzt.

In der Initialisierungsschleife wird die Mustertabelle für die ASCII-Codes bis 158 angelegt. Die Obergrenze kann willkürlich nach oben gesetzt werden. Aus Geschwindigkeitsgründen habe ich die (selten auftretenden) Zeichen von 158 bis 255 nicht eingeschlossen.

```
          "+STR$(15+add_y%),2," « |Okay| » ",d%
          SELECT d%
          CASE 1
            add_y%=MAX(add_y%-1,0)
          CASE 3
            add_y%=MIN(add_y%+1,5)
          ENDSELECT
        UNTIL d%=2
      ELSE IF d%=3
        snap!=TRUE                  ! Snapshot-Utility
      ENDIF
    ENDSELECT
    DEFMOUSE 3
    ' Rubberband darstellen
    GET 400,0,639,20,backgr$
    TEXT 450,15,"Bereich wählen"
    REPEAT
      MOUSE xa%,ya%,mk%
    UNTIL mk%<>0
    GRAPHMODE 3
    nxa%=xa%
    nya%=ya%
    REPEAT
      MOUSE xe%,ye%,mk%
      IF ((xe%<>nxa%)+(ye%<>nya%))*(xe%>xa%)*
         (ye%>ya%)
        BOX xa%,ya%,xe%,ye%
        BOX xa%,ya%,nxa%,nya%
        nxa%=xe%
        nya%=ye%
      ENDIF
    UNTIL mk%=0
    '
    BOX xa%,ya%,xe%,ye%
    IF snap!=TRUE                       ! Snapshot-Utility
      GET xa%,ya%,xe%,ye%,graf$
      HIDEM
      f$="HRDCPY"
      vernr%=1
      WHILE EXIST(f$+STR$(vernr%)+".OBJ") ! und die
                                            Früchte der
        INC vernr%                      ! Arbeit verewigen
      WEND
      OPEN "O",#1,f$+STR$(vernr%)+".OBJ"
      PRINT #1,graf$;
      CLOSE #1
      SHOWM
    ELSE
      GRAPHMODE 0
      BOX 600,2,632,18
      HIDEM
      ' Textanfang suchen (dauert etwa 1-10 sec.)
      TEXT 450,15,"Suche Textanfang"
      ya%=MAX(ya%-3,0)
      FOR x_ofs%=xa% TO xa%+15+add_y%
        PBOX 616+(xa%-x_ofs%),2,616-(xa%-x_ofs%),18
        FOR y_ofs%=ya% TO ya%+15+add_y%
          GET x_ofs%,y_ofs%,x_ofs%+6,y_ofs%+15,char$
          char%=INSTR(charset$,char$) DIV 38
                                  ! bekanntes Textmuster?
          EXIT IF char%<>0 AND char%<>32
                                  ! Text gefunden
        NEXT y_ofs%
        EXIT IF char%<>0 AND char%<>32   ! und raus
                                             hier ...
      NEXT x_ofs%
      start_x%=x_ofs%                       ! Textanfang
      TEXT 450,15,"       Scanne Text"     ! jetzt
                                             geht's los
      DEFFILL 1,0
      PBOX 600,2,632,18
      DEFFILL 1,2,8
      REPEAT
        REPEAT
          PBOX 617+box_wide,2,616-box_wide,18
                                   ! eine Zeile scannen
          GET x_ofs%,y_ofs%,x_ofs%+6,y_ofs%+15,char$
          char%=INSTR(charset$,char$) DIV 38
                                   ! Zeichen bekannt?
          IF char%>0
            INC anz_char%
            txt$=txt$+CHR$(char%)   ! Zeichen merken
          ELSE
```

→



Zeitbedarf

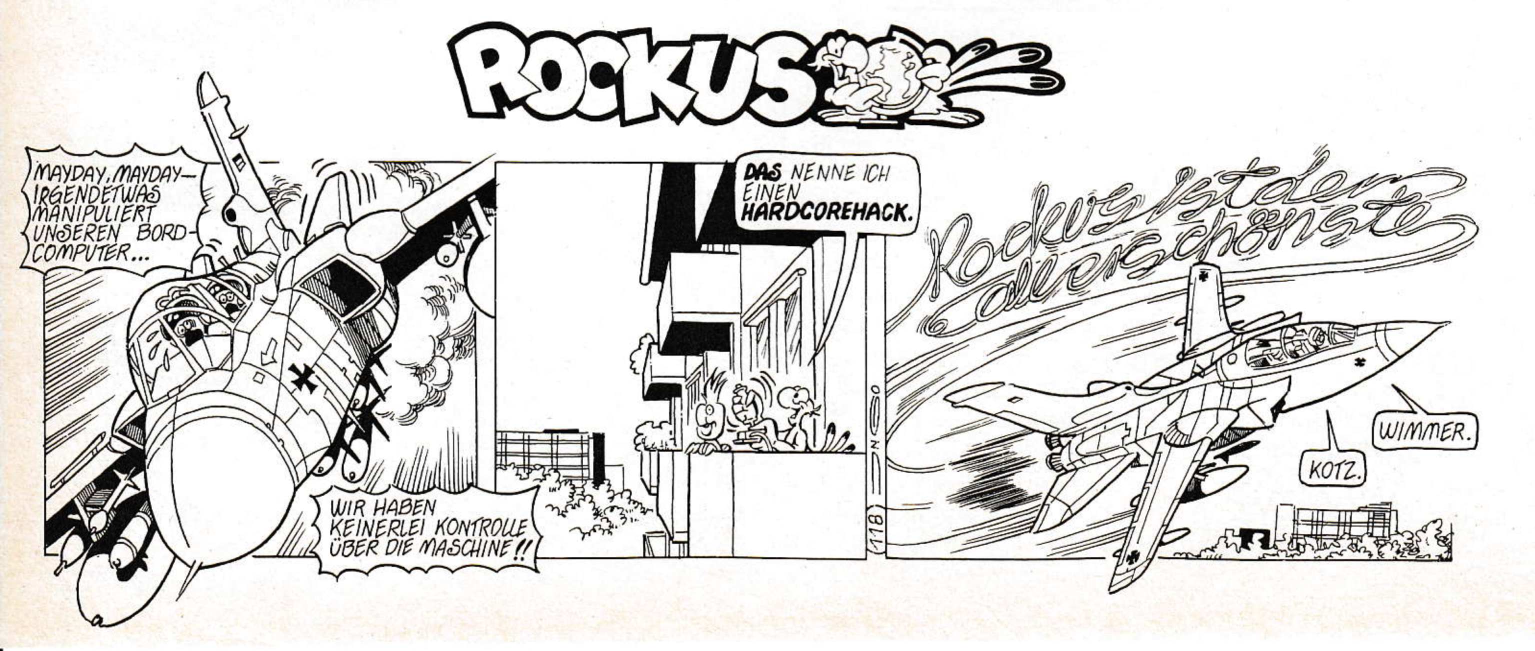
Obwohl das Programm in BASIC geschrieben ist, ist es erstaunlich schnell. Für den Scan-Vorgang werden (je nach Bildgröße) etwa 10-60 Sekunden benötigt. Hierfür entfallen etwa 10 Sekunden auf die Suche nach dem Textanfang. Für eine Scan-Zeile (etwa 60-70 Zeichen) benötigt das Programm jeweils 5-10 Sekunden, so daß keine längeren Wartezeiten auftreten. Rechnet man die Zeitersparnis beim Drukken mit (Texte werden wesentlich schneller gedruckt als Grafik), so dauert die Texthardcopy nicht länger als normaler Grafikdruck. Die Qualität ist aber wesentlich höher (eben NLQ).

Bedienung

Texthardcopy läuft sowohl als Accessory (in THARDCOP.ACC umbennen) als auch als Programm (THARDCOP.PRG). Das Accessory kann durch Anwählen in der Menüleiste bzw. durch Drücken und Halten der beiden Shift-Tasten für ca. 1 Sekunde gestartet werden. Das Programm kann z.B. von einem CLI gestartet werden, wenn man sich in der TOS-Umgebung bewegt. Die verschiedenen Möglichkeiten sind im Listing nochmal ausführlich dokumentiert. Nach dem Programmstart erscheint zunächst eine Dialogbox, in der die Art des Hardcopy-Textes gewählt werden kann (s.o.). Zusätzlich steht ein „HELP"-Button zur Verfügung. Über diesen kann in das Help/Parameter-Menü verzweigt werden. Dort wird ein kurzer Hilfstext ausgegeben. Zusätzlich kann über den Button „Snap" ein Grafikausschnitt auf Diskette gespeichert werden (Snapshot-Utility). Über den Zeilenabstand-Button können exotische Zeilenabstände gewählt werden. Dieser Button wird bisher nicht benötigt und ist für die Zukunft reserviert, falls Atari ein anderes Ausgabeformat wählt. Nachdem Sie Ihre Wahl getroffen haben, erscheint die Aufforderung „Bereich wählen". Sie können nun ein Gummiband aufziehen, das den zu scannenden Textbereich umschließt. Das Gummiband sollte so angelegt werden, daß die Ecken jeweils etwa 5 Pixel vom Textrand entfernt sind. Besonders die Wahl der linken oberen Ecke spielt eine große

```
          txt$=txt$+" "
        ENDIF
        ADD x_ofs%,8
      UNTIL x_ofs%>xe%              ! Zeilenende?
      x_ofs%=start_x%
      ADD y_ofs%,15+add_y%          ! neue Zeile
                                      suchen
      ADD box_wide,(15+add_y%)/((ye%-ya%)/(15+
          add_y%))
      txt$=txt$+CHR$(13)+CHR$(10)
    UNTIL y_ofs%>ye%                ! fertig ...
    DEFMOUSE 0
    SHOWM
    PUT 400,0,backgr$         ! Bild restaurieren
    ALERT 2,"Es wurden "+STR$(anz_char%)+"
    Zeichen|erkannt! Wohin damit ?",1,
    "Drucker|Disk|Müll",d%
    IF d%=1
      LPRINT txt$
    ELSE IF d%=2
      f$="HRDCPY"
      vernr%=1
      WHILE EXIST(f$+STR$(vernr%)+".TXT")  ! und
                                     die Früchte der
        INC vernr%                ! Arbeit verewigen
      WEND
      OPEN "O",#1,f$+STR$(vernr%)+".TXT"
      PRINT #1,txt$
      CLOSE #1
    ENDIF
  ENDIF
  CLR backgr$,graf$
RETURN
```

Rolle, da von hier ausgehend der Textanfang gesucht wird. Als Anhalt kann hier Bild 1 dienen. Nun beginnt der eigentliche Scan-Vorgang. Während das Programm den Textanfang sucht (Meldung „Suche Textanfang"), erscheint in der rechten oberen Ecke ein wachsender Balken. Ist der Textanfang gefunden, wird der Text Zeile für Zeile gescannt (Meldung „Scanne Text"). Der wachsende Balken zeigt auch hier das Voranschreiten der Arbeit an. Wenn der Ausschnitt komplett in ASCII-Text gewandelt ist, werden Sie über den Erfolg des Scannens informiert (Meldung „xxx Zeichen erkannt"). Sie können den fertigen Text nun entweder auf den Drucker ausgeben lassen, verwerfen oder auf Diskette speichern. Der Text wird unter dem File-Namen HRDCOPxx.TXT im aktuellen Directory gespeichert und kann weiterverwendet werden.



Parkwächter

Bildschirmschoner auf Software-Basis gehören heute wohl zum Computer-Alltag. Wird eine gewisse Zeit keine Aktion am Rechner vorgenommen, so sorgt ein im Hintergrund laufendes Programm dafür, daß der Bildschirm dunkel geschaltet wird. Auf diese Weise wird ein Einbrennen des Bildinhalts in die Bildröhre verhindert. Aber nicht nur der Bildschirm läßt sich schonen, sondern auch die Festplatte. Hierzu muß lediglich dafür gesorgt werden, daß diese eine gewisse Zeit nach dem letzten Zugriff automatisch geparkt wird.



Schoner für die Festplatte

Diese Idee ist nicht einmal neu. So sorgte bereits das Betriebssystem der Festplatten der Firma Vortex vor einigen Jahren dafür, daß der Motor der Platten nach einer bestimmten Zeit abgeschaltet wurde. Diese Lösung war natürlich plattenspezifisch. Seit der Einführung des XHDI-Protokolls läßt sich so etwas nun für jede Platte realisieren, vorausgesetzt man verwendet einen Festplattentreiber, der die XHDI-Spezifikation (e**X**tended **H**ard**D**isk **I**nterface) unterstützt.

Das XHDI-Protokoll

Neben dem AHDI-Festplattentreiber von Atari erfreuen sich Treiber von Fremdherstellern schon seit einiger Zeit großer Beliebtheit. Dies liegt in erster Linie daran, daß AHDI bis auf die eigentlichen Treiberfunktionen keinen zusätzlichen Komfort bietet. Dabei ist es sehr praktisch, mit einem Treiber zu arbeiten, der beispielsweise das Booten von beliebigen Partitionen erlaubt oder andere Konfigurationsmöglichkeiten bietet.

Nimmt man die derzeit erhältlichen Festplattentreiber für den Atari unter die Lupe, stellt man fest, daß viele der zusätzlichen Features recht ähnlich sind. Angesprochen wurden diese bisher allerdings durch unterschiedliche Schnittstellen, da keine Normierung des Funktionsumfangs existierte. Die XHDI-Spezifikation stellt nun eine einheitliche Schnittstelle zur Verfügung, die es erlaubt, eine Reihe plattenspezifischer Funktionen unabhängig vom verwendeten Treiber aufzurufen. Welche Funktionen sind das nun im einzelnen? Hier ein kurzer Überblick:

UWORD XHGetVersion(void);
Liefert die Versionsnummer des unterstützten XHDI-Protokolls zurück. Das High-Byte enthält die Version, das Low-Byte die Revision. $0123 stünde somit für Version 1.23. Die im folgenden aufgeführten Funktionen entsprechen XHDI V1.00.

LONG XHInqTarget(UWORD major, UWORD minor, ULING *blocksize, ULONG *device_flags, char *product_name);
Informationen über ein Gerät erfragen. *block_size* enthält eine Angabe über die physikalische Blockgröße auf dem spezifizierten Gerät. *device_flags* liefert einen Attributvektor, *product_name* die Produktbezeichnung des angegebenen Gerätes

.

device_flags ist wie folgt organisiert:

- Bit 0: Gerät kann gestoppt werden
- Bit 1: Gerät hat wechselbare Medien
- Bit 2: Auswurfmechanismus des Gerätes kann verriegelt werden
- Bit 3: Medium kann per Kommando ausgeworfen werden
- Bit 31: Gerät ist zur Zeit reserviert

LONG XHReserve(UWORD major, UWORD minor, UWORD do_reserve, UWORD key);
Gerät reservieren (*do_reserve=1*) oder freigeben (*do_reserve=0*). In manchen Fällen kann es sinnvoll sein, daß ein Programm die XHDI-Funktionen für sich beansprucht und anderen Programmen den Zugriff verwehrt. Für reservierte Geräte sind XHDI-Aufrufe nur dann erlaubt, wenn der von *XHReserve* beim Reservieren zurückgelieferte Zugriffsschlüssel angegeben wird. Dieser Schlüssel wird auch dann benötigt, wenn ein Gerät wieder freigegeben werden soll.

LONG XHLock(UWORD major, UWORD minor, UWORD do_lock, UWORD key);
Auswurfknopf einer Platte verriegeln (*do_lock=1*) oder entriegeln (*do_lock=0*). Dieser Befehl ist für Wechselplatten gedacht. Bei verriegeltem Auswurfknopf ist ein Entnehmen des Mediums nicht möglich. Virtuelle Speicherverwaltungen sollten diese Funktion dann nutzen, wenn sich die Swap-Partition auf einer Wechselplatte befindet.

LONG XHStop(UWORD major, UWORD minor, UWORD do_stop, UWORD key);
Platten parken (*do_stop=1*) oder starten (*do_stop=0*).

LONG XHEject(UWORD major, UWORD minor, UWORD do_eject, UWORD key);
Diese Funktion wirft ein Medium aus (*do_eject=1*) oder zieht es ein (*do_eject=0*). Bisher sind zwar noch keine Platten bekannt, die diese Funktion unterstützen, aber das kann sich ja in Zukunft noch ändern.

ULONG XHDrvMap(void);
Liefert einen Bit-Vektor zurück, der die Laufwerke beschreibt, die vom XHDI-Protokoll unterstützt werden. Der Aufbau dieses Vektors ist analog dem der GEMDOS-Funktion *Drvmap()*.

LONG XHInqDev (UWORD bios_device, *UWORD major, UWORD *minor, ULONG *start_sector, BPB *bpb);
Es werden Major Device Number, Minor Device Number, Startsektor und BPB des angegebenen BIOS-Gerätes zurückgeliefert. Dabei wird im Gegensatz zum BIOS-Aufruf GETBPB der Media-Change-Status nicht zurückgesetzt.

LONG XHInqDriver (UWORD bios_device, char *name, char *version,char *company, UWORD *ahdi_version, UWORD *maxIPL);
Diese Funktion gibt treiberspezifische Informationen zurück.

name:	Zeiger auf Zeichenkette mit Treibernamen
version:	Zeiger auf Zeichenkette mit Versionsnummer des Treibers
company:	Zeiger auf Zeichenkette mit Namen des Herstellers
ahdi_version:	AHDI-Versions-Level
maxIPL:	höchstes IPL (Interrupt Priority Level), unter dem der Treiber für das angegebene Gerät arbeitsfähig ist

Für die Beschreibung der XHDI-Aufrufe gilt folgende Terminologie:

major: Major Device Number

0..7:	Atari-konforme Platte am ACSI-Bus
8..15:	Platten am SCSI-Bus
16..63:	reserviert für Erweiterungen
64:	Gerät am Floppy-Controller
65..255:	für eigene Erweiterungen

minor: Minor Device Number (LUN des ACSI- oder SCSI-Gerätes)

key: Entweder ein 16-Bit-Schlüssel, ermittelt von *XHReserve()* oder 0 bei nicht reserviertem Gerät oder unbekanntem Schlüssel

Wer sich detailliert über den Funktionsumfang des XHDI-Protokolls erkundigen will, findet die offizielle Spezifikation und Beispieldateien auf der Diskette zur vorliegenden Ausgabe der ST- Computer, in diversen Mailboxen [1] sowie eine ausführlichere Darstellung in [2].

Nun stellt sich natürlich noch die Frage, welche Treiber die XHDI-Spezifikation zur Zeit unterstützen. Hier wären HDDRIVER V2.5 (im Lieferumfang von DISKUS und OUTSIDE enthalten), HuSHI V3.0 (Bestandteil der SCSI-Tools von Hard&Soft) sowie zukünftige CBHD-Versionen (gehört zur Scheibenkleister-Software) zu nennen. Weitere Festplattenhersteller haben angekündigt, auch in ihren Treibern das XHDI-Protokoll zu implementieren, so daß für eine breite Basis gesorgt sein dürfte.

XHDI-Anwendung

Aus dem XHDI-Überblick ist ersichtlich, daß mit *XHStop()* eine Funktion zum Parken von Platten zur Verfügung steht. Das AUTOPARK-Accessory schickt diesen Aufruf an alle Geräte, die von einem XHDI-Treiber verwaltet werden, sofern über einen gewissen Zeitraum kein Zugriff erfolgt ist. Welche Geräte in Frage kommen, läßt sich durch *XHDrvMap()* ermitteln. Dabei muß es sich nicht zwangsweise um Festplatten handeln, hier ist das XHDI-Protokoll flexibel. Der zuständige Treiber entscheidet, ob das jeweilige Gerät geparkt werden kann.

Da sich die XHDI-Spezifikation in erster Linie auf physikalische Geräte und nicht auf logische Laufwerke bezieht, nimmt AUTOPARK mit *XHInqDev()* eine Umrechnung der BIOS-Device-Nummern auf die physikalischen Gerätenummern vor. Alle per XHDI erreichbaren Geräte werden zunächst in einer Liste zusammengestellt. Anschließend werden diejenigen Geräte gestrichen, auf deren logische Laufwerke innerhalb des vorgegebenen Zeitrahmens ein Zugriff erfolgt ist. Die restlichen Geräte werden geparkt. Um alle angeschlossenen Platten sofort zu parken, kann der Knopf „Parken“ angeklickt werden. Wird auf ein geparktes Gerät zugegriffen, sorgt der XHDI-Treiber automatisch dafür, daß dieses auch wieder korrekt entparkt wird.

Die Zeit, die vom letzten Zugriff bis zum Parken verstreichen muß, kann dem AUTOPARK-Accessory auf zwei Wegen mitgeteilt werden. Entweder man benutzt den Accessory-Dialog oder verwendet eine Parameterdatei mit dem Namen „AUTOPARK.INF“, die sich in der Regel auf Laufwerk C befinden muß. Diese Datei enthält die Zeitangabe in Minuten, wobei maximal 3 Stellen ausgewertet werden.

Aller guten Dinge sind zwei

Manch einer mag sich fragen, warum es sich bei AUTOPARK um zwei Programme und nicht ausschließlich um ein Accessory handelt. Auf den ersten Blick hätte man die Funktionalität sicherlich auch in einer einzigen Programmdatei unterbringen können. Man muß allerdings berücksichtigen, daß AUTOPARK einen Systemvektor, nämlich *hdv_rw*, verbiegt. Das Verbiegen von Vektoren ist jedoch nur residenten Programmen erlaubt, die nicht mehr aus dem Speicher entfernt werden. Accessories sind zwar resident, aber nur bis zum nächsten Auflösungswechsel. Dann nämlich werden alle Accessories aus dem Speicher entfernt und in der geänderten Auflösung erneut geladen. Wurde nun zwischenzeitlich ein Systemvektor verändert, zeigt dieser anschließend ins Leere, da das Accessory keine Gelegenheit hat, den alten Inhalt vor dem Auflösungswechsel wiederherzustellen. Wer also innerhalb eines Accessories Vektoren verbiegt, hat den nächsten Absturz schon vorprogrammiert. Diejenigen, die das EDISON-Utility auf dem TT mit Farbbildschirm einsetzen, können ein Lied davon singen.

Um *hdv_rw* gefahrlos verbiegen zu können, wird also die PRG-Datei von AUTOPARK benötigt. Diese bleibt auch bei einem Auflösungswechsel im Speicher. Über

den cookie jar wird die Adresse der Tabelle mit den letzten Zugriffszeiten an das Accessory weitergereicht.
Festplatten können beim Parken übrigens durchaus unterschiedlich reagieren. So bewegen manche Modelle lediglich ihren Kopf auf die Parkspur, andere wiederum schalten zusätzlich den Motor ab. Wechselplatten bremsen die Cartridge ab, wie es auch beim Betätigen des Auswurfknopfes der Fall ist.

Bitte beachten

Nicht geparkt werden kann eine Platte von AUTOPARK selbstverständlich dann, wenn das Laufwerk von einem anderen Programm, das ebenfalls das XHDI-Protokoll nutzt, mittels *XHReserve()* reserviert wurde. Dies kann beispielsweise bei den virtuellen Speichermanagern OUTSIDE und VRAM der Fall sein. Zum Arbeiten unter MultiTOS muß bei AUTOPARK.PRG das Global-Bit gesetzt werden, damit das Accessory auch bei aktiver Memory Protection ohne Busfehler auf die Datenstruktur des USPK-cookies zugreifen kann. Ferner muß sich hier die Datei AUTOPARK.INF nicht unbedingt auf Laufwerk C befinden, sondern es gelten die Pfadeinstellungen der GEM-Konfigurationsdatei.

Die Assembler-Direktiven „loadfast“, „ttmem“ und „ttram“ sind für den EASY RIDER-Assembler gedacht und müssen bei der Verwendung eines anderen Assemblers entfernt werden.

US

Literatur:

[1] Beispielsweise Maus MS2 oder ftp.uni-muenster.de in /pub/atari/docs, jeweils abgelegt unter XHDI-100.ZOO

[2] Julian F. Reschke, „Die XHDI-Spezifikation“, ST-Magazin 6/92

```
**************************************
*                                    *
* XHDI-Autoparker V1.00 Accessory    *
*                                    *
* by Uwe Seimet                      *
*                                    *
* (c) 1992 MAXON Computer            *
*                                    *
**************************************
DIALOG    = 0
PARKTIME  = 3
OK        = 6
PARK      = 7
ABORT     = 8
XHStop    = 4
XHDrvMap  = 6
XHInqDev  = 7
GEMDOS    = 1
SUPER     = 32
FOPEN     = 61
FCLOSE    = 62
FREAD     = 63
XBIOS     = 14
SUPEXEC   = 38
APPL_INIT      = 10
EVNT_MULTI     = 25
MENU_REGISTER  = 35
OBJC_DRAW      = 42
OBJC_CHANGE    = 47
FORM_DO        = 50
FORM_DIAL      = 51
FORM_CENTER    = 54
GRAF_MOUSE     = 78
WIND_UPDATE    = 107
RSRC_OBFIX     = 114
END_MCTRL = 2
BEG_MCTRL = 3
_hz_200    = $4ba
_p_cookies = $5a0
XHDIMAGIC= $27011992
         loadfast
         ttmem
         ttram
         text
         lea stack+400,sp
         lea intin,a5                 ;Pointer auf
                                      ;INTIN-Array
         lea intout,a6                ;Pointer auf
                                      ;INTOUT-Array
         moveq #APPL_INIT,d0
         move.l #$00010000,d1
         bsr aes
         move (a6),(a5)               ;apid_nr
         moveq #MENU_REGISTER,d0
         move.l #$01010100,d1         ;Name in
         lea entry,a0                 ;Menüleiste
         bsr aesobj                   ;eintragen
         move (a6),d5                 ;Nummer des
                                      ;Eintrags
                                      ;merken
         bmi quit
         move #_objcnr,(a5)
         bra.s fix
obfix:   moveq #RSRC_OBFIX,d0
         move.l #$01010100,d1
         lea objc000,a0               ;Objektdaten
         bsr aesobj                   ;umrechnen
fix:     subq #1,(a5)
         bpl obfix
         clr -(sp)
         pea parname
         move #FOPEN,-(sp)
         trap #GEMDOS
         addq.l #8,sp
         move.l d0,d3                 ;keine
         bmi.s loop                   ;Parameter-
         pea parbuff
         pea 3
         move d3,-(sp)
         move #FREAD,-(sp)
         trap #GEMDOS
         lea 10(sp),sp
         move d3,-(sp)
         move #FCLOSE,-(sp)
         trap #GEMDOS
         addq.l #4,sp
         lea parbuff,a0
         bsr get
loop:    move #$10,(a5)               ;MU_MESAG
         tst time
         beq.s notime
         move #$30,(a5)               ;MU_MESAG|
                                      ;MU_TIMER
         move.l #$ea600000,28(a5)     ;1 min warten
notime:  lea ev_buff,a0               ;Buffer für
                                      ;GEM-Messages
         moveq #EVNT_MULTI,d0
         move.l #$10070100,d1
         bsr aesobj
         clr.l -(sp)
         move #SUPER,-(sp)
         trap #GEMDOS
         addq.l #6,sp
```

→

```
              move.l d0,-(sp)          ;alter
                                       ;Stackpointer
              move.l _p_cookies,d0     ;keine
              beq supret               ;cookies-
              sub.l a3,a3
              sub.l a4,a4
              move.l d0,a0
fxhdi:        movem.l (a0)+,d0/a1
              tst.l d0
              beq endjar
              cmp.l #"XHDI",d0
              bne.s noxhdi
              cmp.l #XHDIMAGIC,-4(a1)
              bne.s noxhdi
              move.l a1,a4
              bra fxhdi
noxhdi:       cmp.l #"USPK",d0
              bne fxhdi
              move.l a1,a3
              bra fxhdi
endjar:       move.l a4,d0             ;kein
              beq supret               ;XHDI-Treiber-
              move.l a3,d0             ;AUTOPARK.PRG
              beq supret               ;nicht
                                       ;installiert-
              btst #5,1(a6)            ;MU_TIMER?
              beq supret               ;nein-
              move #XHDrvMap,-(sp)     ;XHDI-
              jsr (a4)                 ;Gerätevektor
              addq.l #2,sp             ;holen
              move.l d0,d6
*Liste aller XHDI-Devices erstellen
              lea devices,a2           ;Flag für
              move.l #-1,(a2)          ;Tabellenende
              moveq #0,d7
inquire:      bsr inqdev
              bmi.s inqnext
              move.l major,(a2)+
              move.l #-1,(a2)
inqnext:      addq #1,d7
              cmp #32,d7
              bne inquire
*Totzeit überprüfen
              moveq #0,d7
test:         bsr inqdev
              bmi.s next
              move d7,d0
              add d0,d0
              add d0,d0
              move time,d1             ;Zeit in
              mulu #12000,d1           ;200 Hertz-
                                       ;Schritten
              add.l (a3,d0),d1         ;letzte
                                       ;Zugriffszeit
              cmp.l _hz_200,d1
              bcs.s next               ;parken-
              move.l major,d0
              lea devices,a0
skipdev:      cmp.l #-1,(a0)
              beq.s next               ;Listenende
              cmp.l (a0)+,d0
              bne skipdev
              move.l #-2,-4(a0)        ;aus Liste
              bra skipdev              ;streichen
next:         addq #1,d7
              cmp #32,d7
              bne test
              lea devices,a2
park:         cmp.l #-1,(a2)
              beq.s supret
              cmp.l #-2,(a2)+
              beq park
              move.l -4(a2),major
              clr -(sp)                ;Dummy-Key
              move #1,-(sp)            ;Platte parken
              move minor,-(sp)
              move major,-(sp)
              move #XHStop,-(sp)
              jsr (a4)
              lea 10(sp),sp
              bra park
supret:       move #SUPER,-(sp)
              trap #GEMDOS
              addq.l #6,sp
message:      btst #4,1(a6)            ;MU_MESAG?
              beq loop                 ;nein-
              lea ev_buff,a0
              cmp #40,(a0)             ;AC_OPEN?
              bne loop                 ;nein-
              cmp 8(a0),d5             ;AUTOPARK?
              bne loop                 ;nein-
              moveq #BEG_MCTRL,d0
              bsr update
              bsr.s dialog
              moveq #END_MCTRL,d0
              bsr update
              bra loop
quit:
              moveq #19,d0             ;appl_exit
              move.l #$00010000,d1
              bsr aes
              clr -(sp)
              trap #GEMDOS             ;das war's
dialog:
              lea objc000,a2
              move #PARKTIME*24,d0
              move.l 12(a2,d0),a0
              move.l (a0),a0
              moveq #0,d0
              move time,d0
              bsr int
              moveq #FORM_CENTER,d0
              move.l #$00050100,d1
              move.l a2,a0
              bsr aesobj
              movem.l 2(a6),a3-a4      ;form_xy,
                                       ;form_wh
              clr d2
              movem.l a3-a4,2(a5)
              movem.l a3-a4,10(a5)
              bsr form_dial
              moveq #OBJC_DRAW,d0
              move.l #$06010100,d1
              move.l #$00000002,(a5)
              move.l 2(a6),4(a5)
              move.l 6(a6),8(a5)
              move.l a2,a0             ;Dialogbox
              bsr aesobj               ;darstellen
              moveq #FORM_DO,d0
              move.l #$01010101,d1
              move #PARKTIME,(a5)      ;Eingabefeld
              move.l a2,a0             ;Dialog
              bsr aesobj               ;starten
              move (a6),d3
              bclr #15,d3
              cmp #ABORT,d3            ;Abbruch?
              beq.s abort              ;ja-
              cmp #PARK,d3             ;alle Geräte
              beq.s parkall            ;parken-
              move #PARKTIME*24,d0
              move.l 12(a2,d0),a0
              move.l (a0),a0
              bsr get
abort:        moveq #3,d2
              movem.l a3-a4,2(a5)
              bsr fo_dial
              moveq #OBJC_CHANGE,d0
              move.l #$08010100,d1
              move.l a5,a0
              move d3,(a0)+
              clr (a0)+
              movem.l a3-a4,(a0)
              clr.l 12(a5)
              move.l a2,a0             ;Exit-Button
              bra aesobj               ;deselektieren
parkall:
              move #2,(a5)             ;Biene als
              moveq #GRAF_MOUSE,d0     ;Mauscursor
              move.l #$01010100,d1
              bsr aes
              pea stop(pc)
              move #SUPEXEC,-(sp)
              trap #XBIOS
```

→

```
                addq.l #6,sp
                clr (a5)                        ;Pfeil als
                moveq #GRAF_MOUSE,d0            ;Mauscursor
                move.l #$01010100,d1
                bsr aes
                bra abort

stop:
                move.l _p_cookies,d0            ;keine
                beq.s error                     ;cookies-
                move.l d0,a0
nxhdi:          movem.l (a0)+,d0/a4
                tst.l d0
                beq.s error
                cmp.l #"XHDI",d0
                bne.s nxhdi
                cmp.l #XHDIMAGIC,-4(a4)
                bne.s nxhdi

                move #XHDrvMap,-(sp)            ;XHDI-
                jsr (a4)                        ;Gerätevektor
                addq.l #2,sp                    ;holen
                move.l d0,d6
                moveq #0,d7
nxt:            bsr inqdev
                bmi.s nxtdev
                clr -(sp)                       ;Dummy-Key
                move #1,-(sp)                   ;Platte parken
                move minor,-(sp)
                move major,-(sp)
                move #XHStop,-(sp)
                jsr (a4)
                lea 10(sp),sp
nxtdev:         addq #1,d7
                cmp #32,d7
                bne nxt
error:          rts


fo_dial:
                movem.l a3-a4,10(a5)
form_dial:
                moveq #FORM_DIAL,d0
                move.l #$09010100,d1
                move d2,(a5)
aesobj:
                move.l a0,addrin
aes:            lea contrl,a0
                move d0,(a0)
                movep.l d1,3(a0)
                move.l #aespb,d1
                move #$c8,d0
                trap #2
                rts

update:
                move d0,(a5)
                moveq #WIND_UPDATE,d0
                move.l #$01010000,d1
                bra aes

*Gerätenummer zu logischem Laufwerk erfragen
inqdev:
                moveq #-1,d0
                btst d7,d6                      ;kein
                beq.s nodev                     ;XHDI-Device-
                clr.l -(sp)
                clr.l -(sp)
                pea minor
                pea major
                move d7,-(sp)
                move #XHInqDev,-(sp)            ;Gerätenummern
                jsr (a4)                        ;holen
                lea 20(sp),sp
nodev:          tst d0
                rts

*Umwandlung HEX in ASCII
int:
                tst.l d0
                beq int5                        ;Null-
                moveq #2,d3
                move.l a0,a1
                move.l #100,d2  ;Startwert für
                                 Subtraktion
int1:           moveq #-1,d1    ;zählt Subtraktionen
int0:           addq.b #1,d1
                sub.l d2,d0     ;so oft wie möglich
                                 subtrahieren
                bcc int0
                add.l d2,d0
                divu #10,d2     ;nächste Stelle
                tst.b d1
                bne int3
                cmp.l a1,a0
                beq int4
int3:           add.b #"0",d1                   ;Ziffer nach
                move.b d1,(a0)+                 ;ASCII wandeln
int4:           dbra d3,int1
                clr.b (a0)
                rts
int5:           move.b #"0",(a0)+
                clr.b (a0)
                rts

*Umwandlung ASCII in HEX
get:
                moveq #0,d0         ;Ergebnis löschen
                moveq #0,d1
getloop:        move.b (a0)+,d1     ;Ziffer holen
                beq.s getret        ;Ende-
                sub.b #'0',d1       ;in HEX wandeln
                add.l d0,d0
                move.l d0,-(sp)
                asl.l #2,d0
                add.l (sp)+,d0
                add.l d1,d0
                bra getloop
getret:         move d0,time
                rts


                data

aespb:          dc.l contrl,global
                dc.l intin,intout
                dc.l addrin,addrout

entry:          dc.b "  XHDI-Autoparker",0

parname:        dc.b "AUTOPARK.INF",0

                even

G_BOX     = 20
G_TEXT    = 21
G_BUTTON  = 26
G_FTEXT   = 29

objc000:        dc.w $ffff
                dc.w $0001,$0008
                dc.w G_BOX
                dc.w $0000,$0010
                dc.l $00021100
                dc.w $0000,$0000
                dc.w $0027,$000d

                dc.w $0002
                dc.w $ffff,$ffff
                dc.w G_BUTTON
                dc.w $0001,$0011
                dc.l spec000
                dc.w $0008,$0001
                dc.w $0017,$0801

                dc.w $0003
                dc.w $ffff,$ffff
                dc.w G_TEXT
                dc.w $0000,$0000
                dc.l spec001
                dc.w $000b,$0004
                dc.w $0410,$0001

                dc.w $0004
                dc.w $ffff,$ffff
                dc.w G_FTEXT
                dc.w $0008,$0000
                dc.l spec002
                dc.w $0002,$0006
                dc.w $0023,$0001

                dc.w $0005
                dc.w $ffff,$ffff
                dc.w G_TEXT
```

→

```
           dc.w $0000,$0000
           dc.l spec003
           dc.w $0004,$0008
           dc.w $061e,$0001

           dc.w $0006
           dc.w $ffff,$ffff
           dc.w G_TEXT
           dc.w $0000,$0000
           dc.l spec004
           dc.w $000a,$0009
           dc.w $0012,$0001

           dc.w $0007
           dc.w $ffff,$ffff
           dc.w G_BUTTON
           dc.w $0007,$0000
           dc.l spec005
           dc.w $0005,$000b
           dc.w $0009,$0001

           dc.w $0008
           dc.w $ffff,$ffff
           dc.w G_BUTTON
           dc.w $0005,$0000
           dc.l spec006
           dc.w $000f,$000b
           dc.w $0009,$0001

           dc.w $0000
           dc.w $ffff,$ffff
           dc.w G_BUTTON
           dc.w $0025,$0000
           dc.l spec007
           dc.w $0019,$000b
           dc.w $0009,$0001

spec000:   dc.b "XHDI-Autoparker V1.00",0

spec001:   dc.l txt001,plt001,val001
           dc.w $0005
           dc.w $0006
           dc.w $0000
           dc.w $1180
           dc.w $0000
           dc.w $ffff
           dc.w $0017,$0001
txt001:    dc.b "(C) 1992 by Uwe Seimet",0
plt001:    dc.b 0
val001:    dc.b 0

spec002:   dc.l txt002,plt002,val002
           dc.w $0003
           dc.w $0006
           dc.w $0002
           dc.w $1180
           dc.w $0000
           dc.w $ffff
           dc.w $0004,$0024
txt002:    dc.b "___",0
plt002:    dc.b "Zeit bis zum Parken in Minuten:
                ___",0
val002:    dc.b "999",0

spec003:   dc.l txt003,plt003,val003
           dc.w $0005
           dc.w $0006
           dc.w $0000
           dc.w $1180
           dc.w $0000
           dc.w $ffff
           dc.w $002a,$0001
txt003:    dc.b "XHDI-kompatibler Treiber und
                AUTOPARK.PRG",0
plt003:    dc.b 0
val003:    dc.b 0

spec004:   dc.l txt004,plt004,val004
           dc.w $0005
           dc.w $0006
           dc.w $0000
           dc.w $1180
           dc.w $0000
           dc.w $ffff
           dc.w $0019,$0001
txt004:    dc.b "müssen installiert sein!",0
plt004:    dc.b 0
val004:    dc.b 0

spec005:   dc.b "OK",0

spec006:   dc.b "Parken",0

spec007:   dc.b "Abbruch",0

_objcnr equ 0009


           bss

contrl:    ds.w 11

global:    ds.w 15

intin:     ds.w 64

intout:    ds.w 64

addrin:    ds.w 64

addrout:   ds.w 64

ev_buff:   ds.w 8

ap_buff:   ds.w 8

           even

stack:     ds.l 100

time:      ds.w 1

major:     ds.w 1
minor:     ds.w 1

devices:   ds.w 68

parbuff:   ds.b 4  ;Puffer für Parameterdatei
```

```
***************************************************
*                                                 *
*             XHDI-Autoparker V1.00               *
*                                                 *
*   Parkt Geräte, die von einem XHDI-Treiber      *
*  verwaltet werden, nach einer einstellbaren     *
*               Zeit automatisch.                 *
*                                                 *
* Benötigt werden AUTOPARK.PRG und AUTOPARK.ACC   *
*                                                 *
*                  by Uwe Seimet                  *
*                                                 *
*            (c) 1992 MAXON Computer              *
*                                                 *
***************************************************


GEMDOS   = 1
CCONWS   = 9
SUPER    = 32
PTERMRES = 49
FOPEN    = 61
FCLOSE   = 62
FREAD    = 63
MSHRINK  = 74

XBIOS    = 14
SUPEXEC  = 38

resvalid = $426
resvector= $42a
hdv_rw   = $476
_hz_200  = $4ba
_p_cookies= $5a0

XHDIMAGIC= $27011992
USPK     = $5553504b


   loadfast
```

→

```
        ttmem
        ttram


                text

                move.l 4(sp),a5
                move.l 12(a5),a6                ;TEXT-Segment
                add.l 20(a5),a6                 ;DATA-Segment
                add.l 28(a5),a6                 ;BSS-Segment
                lea $100(a6),a6                 ;für Basepage
                pea (a6)
                pea (a5)
                clr -(sp)
                move #MSHRINK,-(sp)             ;überzähligen
                trap #GEMDOS                    ;Speicher
                lea 12(sp),sp                   ;freigebem

                pea install(pc)
                move #SUPEXEC,-(sp)
                trap #XBIOS
                addq.l #6,sp
                tst.b instflg                   ;bereits
                bne.s quit                      ;installiert-

                pea message1
                move #CCONWS,-(sp)
                trap #GEMDOS
                addq.l #6,sp
                add.l #newcook-lastacst,a6      ;für cookies
                add.l cookmem,a6                ;benötiger
                sub.l 28(a5),a6                 ;Speicher
                clr -(sp)
                pea (a6)                        ;Programm
                move #PTERMRES,-(sp)            ;resident
                trap #GEMDOS                    ;halten

quit:           pea message2
                move #CCONWS,-(sp)
                trap #GEMDOS
                addq.l #6,sp
                clr -(sp)                       ;nicht
                trap #GEMDOS                    ;installieren

install:
                move.l _p_cookies,d0            ;cookie jar
                bne.s jar                       ;vorhanden-

                move.l resvalid,valsave
                move.l resvector,vecsave
                move.l #reset,resvector
                move.l #$31415926,resvalid
                moveq #31,d1                    ;32 cookie-
                                                ;Einträge
                lea newcook,a0
                move.l a0,_p_cookies
                bra.s newjar

jar:            move.l d0,a0
                lea newcook,a1
                moveq #0,d2                     ;zählt cookies
loop:           movem.l (a0)+,d0-d1
                cmp.l #USPK,d0
                seq instflg                     ;schon
                beq.s ret                       ;installiert-
                movem.l d0-d1,(a1)              ;Einträge
                addq.l #8,a1                    ;kopieren
                addq.l #1,d2
                tst.l d0
                bne loop
                subq.l #8,a0                    ;cookie jar
                cmp.l d1,d2                     ;voll?
                bcs.s nofull                    ;nein-
                lea -8(a1),a0                   ;8 zusätzliche
                addq.l #8,d1                    ;Einträge
                move.l #newcook,_p_cookies
newjar:         move.l d1,d2
        addq.l #1,d2
                asl.l #3,d2
                move.l d2,cookmem
nofull:         lea lastacst,a1
                move.l #"USPK",(a0)+
                move.l a1,(a0)+
                movem.l d0-d1,(a0)
                lea lastacst,a0
                moveq #31,d0
init:           move.l _hz_200,(a0)+            ;Timerwerte
                dbra d0,init                    ;zurücksetzen
                move.l hdv_rw,old_rw
                move.l #new_rw,hdv_rw
ret:            rts


                dc.l "XBRA"
                dc.l "USPK"
old_rw:         dc.l 0
new_rw:
                move 14(sp),d0                  ;Drivenummer
                cmp #32,d0                      ;Absicherung
                bcc.s norw                      ;gegen falsche
                                                ;Drivenummer
                add d0,d0
                add d0,d0
                lea lastacst,a0                 ;Zugriffszeit
                move.l _hz_200,(a0,d0)          ;merken
norw:           move.l old_rw(pc),a0
                jmp (a0)

*cookie jar nach Reset zurücksetzen
reset:
                clr.l _p_cookies
                move.l valsave,resvalid
                move.l vecsave,resvector
                jmp (a6)


                data

message1:dc.b $0d,$0a
                dc.b "XHDI-Autoparker V1.00 installiert"
                dc.b $0d,$0a
                dc.b "(c) 1992 by Uwe Seimet",$0d,$0a,$00

message2:dc.b $0d,$0a
                dc.b "XHDI-Autoparker ist bereits "
                dc.b "installiert",$0d,$0a,$00


                bss

lastacst:ds.l 32   ;Tabelle mit Zugriffszeiten

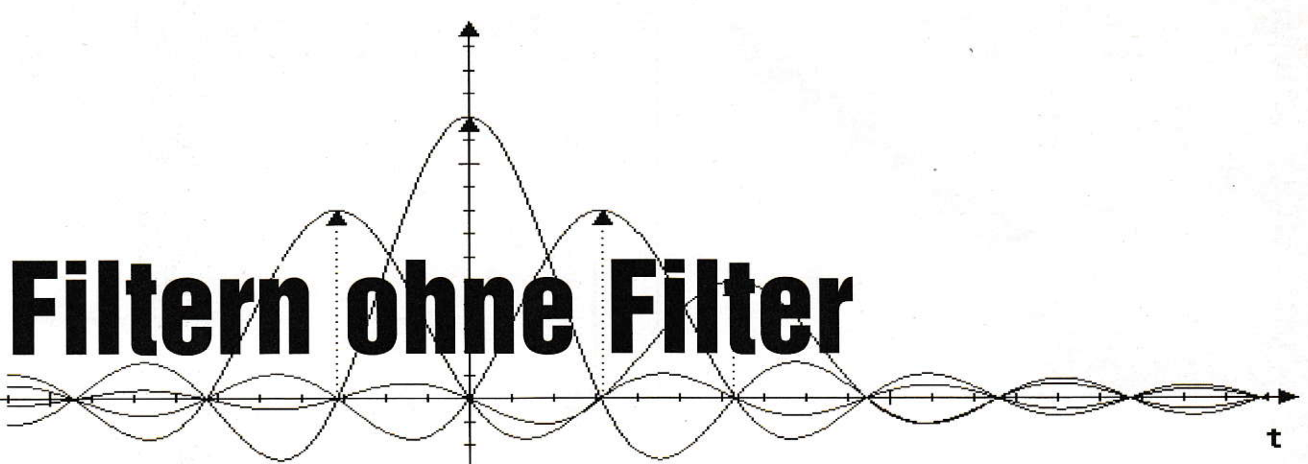
valsave: ds.l 1    ;altes Reset-Magic

vecsave: ds.l 1    ;alter Reset-Vektor

newcook: ds.l 512  ;Platzhalter für 256 Einträge

cookmem: ds.l 1    ;Größe des neuen cookie jar

instflg: ds.b 1    ;Flag für Zweitinstallation
```



Filtern ohne Filter

Vom Oversampling und Tiefpässen

Wer eine Reihe von Meß- oder sonstigen Daten aufgenommen hat und wissen möchte, was zwischen den einzelnen Meßpunkten passiert, muß interpolieren. Eine Möglichkeit, dies zu bewerkstelligen, wird in diesem Beitrag beschrieben. Und wer darüber hinaus schon immer mal wissen wollte, was sich hinter dem Schlagwort Oversampling verbirgt und warum Nachrichtensprecher keine karierten Jacketts tragen, sollte weiterlesen...

Unter Interpolation versteht man bekanntlich die näherungsweise Approximation einer Funktion, von der man nur Stichproben kennt. Diese Stichproben können stündliche Temperaturmessungen, einzelne Punkte einer Bauelementekennlinie oder Amplitudenwerte eines digital abgetasteten Tonsignals sein. Und da Englisch etwas technischer klingt, nennt man die Interpolation im letzteren Falle *Oversampling*.

Die alte Leier

Mein Gott, fällt denen denn nichts Neues ein, wird sich mancher fragen, das kenne ich doch alles schon. Newton, Lagrange, Spline - alles alte Hüte. In der Tat ist die Interpolation so alt wie die moderne Mathematik. Schon der alte Newton hat ein geniales, für Computer sehr geeignetes Verfahren entwickelt, das der eine oder andere vielleicht noch aus der Schule kennt. Es basiert auf der fundamentalen Erkenntnis, daß durch n Punkte immer genau ein Polynom n-1ten Grades hindurchgeht. Das heißt, durch zwei Punkte geht immer eine Gerade, durch drei Punkte eine Parabel usw. Bild 1 zeigt in seiner linken Hälfte ein Polynom 7. Grades, das 8 Punkte verbindet. Aber Polynome sind ja nicht das einzige, was die Welt im Innersten zusammenhält. Kubische Splines würden z.B. den Verlauf der 8 Punkte in Bild 1 viel besser approximieren als das Polynom, aber hier ist der Rechenaufwand höher, und letztlich sind sie ja auch Polynome. Ideal wäre ein Verfahren mit niedrigem Rechenaufwand, das nicht so stark schwingt und auch nicht so zackig aussieht wie ein simpler *V_PLINE*-Aufruf. Der hier vorgestellte Algorithmus verbraucht wenig Rechenzeit und Speicherplatz und, für Programmierer besonders angenehm, er beansprucht wenig Quelltext.

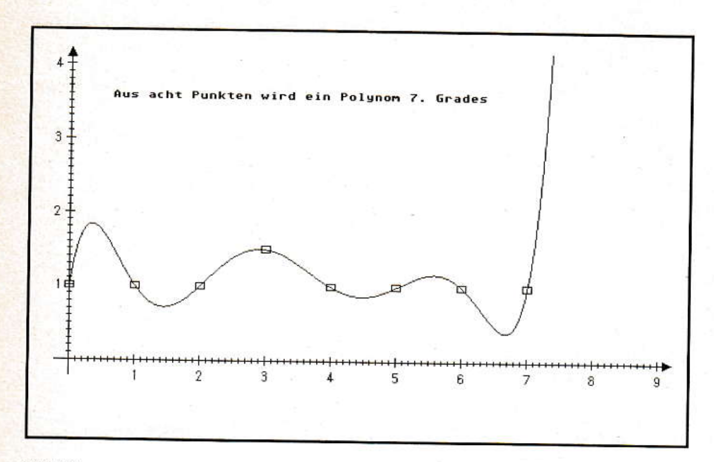
Grau ist alle Theorie

Wollte ein CD-Player mit Oversampling sein Interpolationsproblem mit Polynomen lösen, hätte er bei 44 kHz Abtastfrequenz und 5 Minuten Titellänge ein Polynom 13199999. Grades auszurechnen. Da das auch für den dicksten Signalprozessor zuviel ist, wendet er ein Verfahren an, das ähnlich dem hier beschriebenen funktioniert.

Wie wir aus der Schule wissen, kann man ein periodisches Signal in sogenannte Harmonische, d.h. Grundschwingung und Oberwellen zerlegen (siehe Bild 2). Die Höhe der senkrechten Pfeile in Bild 2 repräsentiert den Betrag des jeweiligen Fourier-Koeffizienten, der seinerseits die Amplitude der n-ten Harmonischen (d.h. Schwingung mit n-facher Grundfrequenz) verkörpert. So ein Pfeil ist in der Signaltheorie eigentlich ein ganz schmales Rechteck, dessen Breite gegen Null und dessen Höhe dafür gegen unendlich geht, so daß es trotz der unendlichen Schmalheit einen Flächeninhalt hat. Da aber das Format der ST-Computer die Darstellung der unendlichen Höhe nicht zuläßt, gibt man mit der Pfeilspitze nur den Betrag des Flächeninhaltes des Rechtecks an, welches in der Signaltheorie *Dirac-Stoß* genannt wird.

In Bild 3 gehen wir zu einer weniger anschaulichen, aber dafür wissenschaftlich exakteren Darstellung über. Die erste Frage, die einem bei diesem Bild durch den Kopf schießt, ist: Wozu die negativen Frequenzen? Nun, wissenschaftlich unexakt erklärt kann man sich das folgendermaßen vorstellen. Wenn man auf der Zeitachse in positiver Richtung entlangläuft, zählt man soundsoviel Perioden einer Harmonischen pro Sekunde, ergo positive Frequenz. Wenn man hingegen in umgekehrter Richtung läuft, zählt man die gleiche Anzahl von Perioden pro negativer Sekunde, ergo negative Frequenz. Da aber beide Aussagen zutreffen, trägt man kurzerhand positive und negative Frequenz auf. Aber auch die Amplituden sind sowohl beim Vorwärts- als auch beim Rückwärtslaufen die gleichen, deshalb ist der Wert bei f immer genau so groß wie bei $-f$.

Wenn wir aus dem Frequenzbild die Zeitfunktion wieder zurückgewinnen wollen, zählen wir einfach den Wert von $+f$ und $-f$ zusammen, um auf den Wert der Amplitude der jeweiligen Harmonischen zu kommen. Und da $+f$ und $-f$ vollkommen gleichberechtigt sind, haben sie jeweils



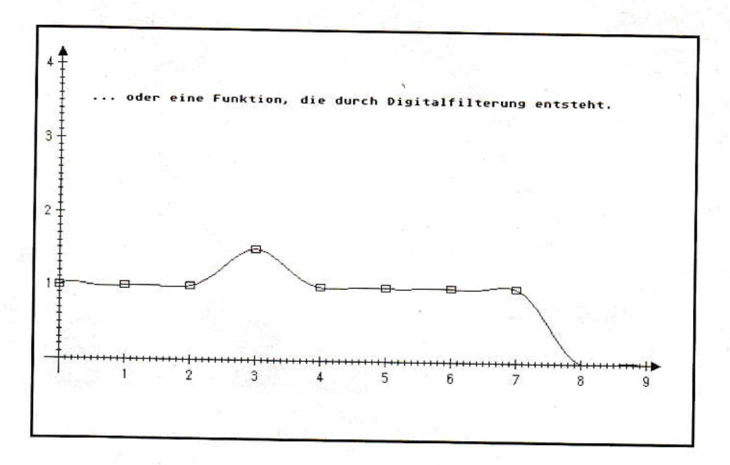


Bild 1

die Hälfte der Amplitude der Harmonischen als Betrag (sie sind halb so groß, wie ihr Äquivalent in Bild 2).

Nu gugge an

Die gepunktete Hüllkurve in Bild 3 stellt die Funktion *sin(x)/x* dar. Sie heißt Spaltfunktion und wird durch *si(x)* abgekürzt. Wie durch ein Wunder liegen alle Fourier-Darstellungen von Rechteckfolgen unabhängig vom Tastverhältnis in dieser Hüllkurve, es ändert sich nur der Abstand der Dirac-Stöße, wenn man die Breite des Einzelimpulses *ti* konstant hält und nur die Periodendauer *T* der Rechteckschwingung ändert. Aber ein Wunder ist das natürlich nicht, sondern Mathematik. Die Hüllkurve ist nämlich das Frequenzspektrum eines einzelnen Impulses aus der Rechteckfolge. Ja richtig, auch nichtperiodische Signale haben ein Frequenzspektrum, wie sollte man sonst einen Rechteckimuls im Telefon hören können, das doch nur Frequenzen von 300 Hz bis 3,4 kHz überträgt. Der Unterschied zur Fourier-Reihe besteht darin, daß das Spektrum eines einzelnen Impulses kontinuierlich ist und somit nicht aus einzelnen Dirac-Stößen besteht. Man könnte sich ein nichtperiodisches Signal z.B. als periodisches mit unendlicher Periodendauer vorstellen. Dadurch wird die Folge der Dirac-Stöße unendlich dicht, und sie werden auch unendlich klein. Wenn ich aber etwas unendlich Hohes wieder unendlich niedrig mache, kommt etwas Endliches heraus, und das ist gerade das Spektrum eines Einzelimpulses (diesmal ohne Pfeil, weil endlich).

Heureka!

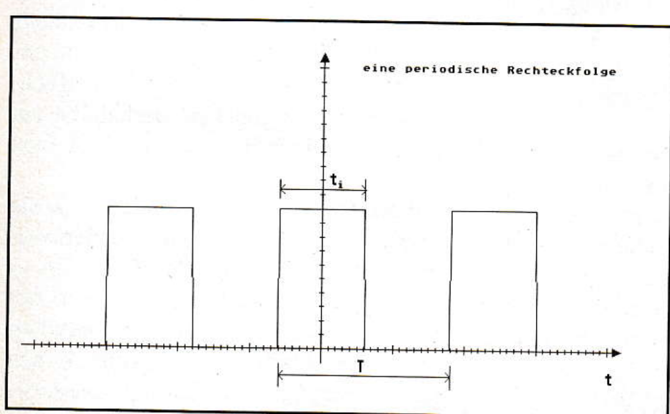
Und hier haben wir auch schon ein fundamentales Grundgesetz der Signaltheorie entdeckt: Periodifizierung eines Impulses im Zeitbereich wird zur Abtastung (und nichts anderes sind Dirac-Stöße, die in einer Hüllkurve liegen) des Spektrums dieses Impulses im Frequenzbereich, wobei der Abstand der Dirac-Stöße genau der Grundfrequenz (1/T) der Periodifizierung entspricht. Wobei wir unter Impuls eine beliebige zeitabhängige Funktion verstehen.

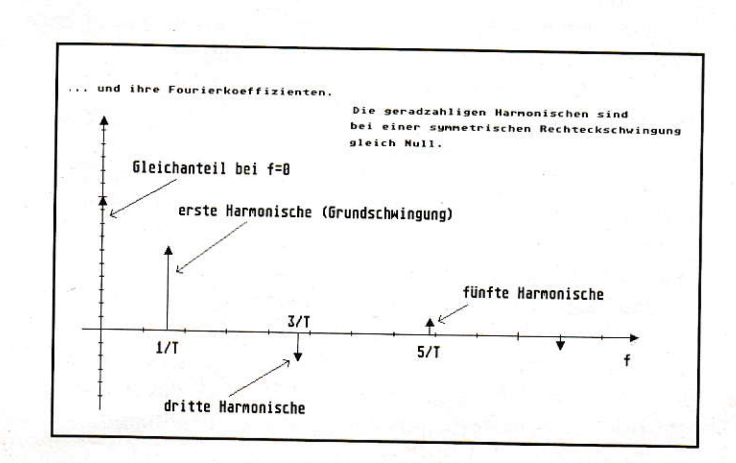
Aber was dem Frequenzbereich recht ist, kann dem Zeitbereich nur billig sein. Denn aus der Mathematik folgt, daß auch Periodifizierung im Frequenzbereich zur Abtastung im Zeitbereich wird, und gerade das ist es ja, was wir in unserem Zeitbereich haben, wenn wir interpolieren: Abtastwerte. Demzufolge müßte das Frequenzspektrum der Abtastwerte eine periodische Folge des Spektrums der Originalfunktion sein (siehe Bild 4), die wir ja mit unserer Interpolation finden wollen. Und das müßte nicht nur so sein, es ist auch so.

Wenn es uns nun also gelänge, das Originalspektrum aus dem periodischen Spektrum herauszufiltern, hätten wir eine eindeutige Repräsentation der Originalfunktion im Frequenzbereich. Doch hier taucht ein Problem auf, das jeder Programmierer von Spielen mit Digisound kennt. Die „Periodendauer" im Frequenzspektrum ist - wer hätte es anders erwartet - der Kehrwert des Abstandes zweier Abtastwerte im Zeitbereich, so wie die Periodendauer im Zeitbereich ja auch der Kehrwert des Abstandes zweier Dirac-Stöße im Frequenzbereich ist (siehe Bild 3). Somit werden die Abstände der Teilfunktionen in Bild 4 mit Größerwerden der Abtastzeit immer geringer.

Sollte die Abtastzeit zwischen zwei Proben (‚Samples') zu groß werden, überlappen sich die einzelnen Teilspektren, und eine eindeutige Identifikation des Spektrums der Originalfunktion ist nicht mehr möglich (siehe Bild 5). Dieses Hineinreichen eines weiter oberhalb liegenden Spektrums in das Originalspektrum hören wir bei schlechtem Digisound als hohes Zwitschern im Hintergrund oder wir sehen es als Moiré auf dem karierten Jackett des Nachrichtensprechers, wenn die „Frequenz" des Musters zu hoch ist für die „Abtastrate" der Fernsehkamera. Nur daß sich in diesem Falle Frequenz und Abtastrate auf den Ort und nicht auf die Zeit beziehen.

Bild 2





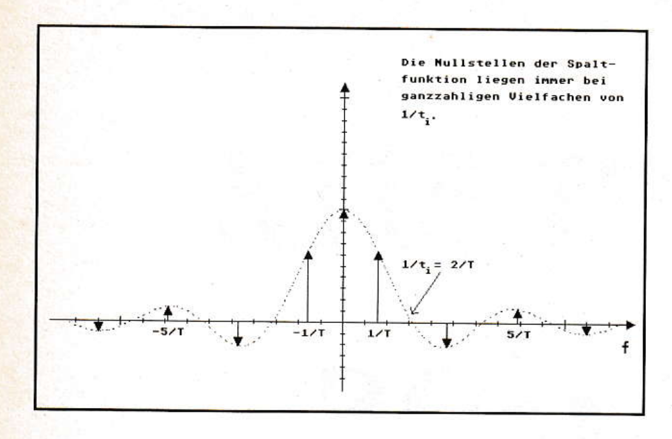


Bild 3

Schreiten wir zur Extraktion ...

Nehmen wir einmal an, die Abtastzeit sei kurz genug, so daß sich die Spektren nicht überlappen. Wenn das der Fall ist, kann man das Originalspektrum sehr leicht herausfiltern, indem man den ganzen Salat mit einer nichtperiodischen Rechteckfunktion multipliziert (Bild 6). Herausfiltern ist dabei wörtlich gemeint, denn so ein Rechteck im Frequenzbereich ist ja nichts anderes als ein idealer Tiefpaß, der unterhalb einer gewissen Frequenz alles durchläßt und oberhalb dieser Frequenz nichts. Jetzt haben wir das Spektrum der Originalfunktion, wie es leibt und lebt, und brauchen, da es eine eindeutige Abbildung derselben ist, nur noch die Rücktransformation vorzunehmen. Aber gerade hier ist der Haken: Die Rücktransformation ist nämlich sehr aufwendig und würde den Rahmen dieses Artikels sprengen. Doch Murphy wird diesmal überlistet, es gibt nämlich eine Regel, die besagt, daß die Zeitfunktion eines Produktes im Frequenzbereich (zwischen Rechtecktiefpaß und periodischer Spektrenfolge) gleich der Faltung der beiden Zeitfunktionen der Faktoren ist (was Faltung ist, kommt gleich). Das heißt, ich nehme die Zeitfunktion des periodischen Spektrums (unsere Abtastfolge, aus der immer noch nichts gescheites geworden ist) und die Zeitfunktion des Filterrechtecks und falte sie miteinander.

Rüber und nüber

Faltung ist leider nicht das einfache Herumklappen irgendeines Funktionsteils auf einen anderen, sondern auch wieder eine komplizierte Operation, die zu erklären den Rahmen des Artikels sprengen würde. Wir haben aber Glück, da die Faltung einer Funktion (der Zeitfunktion des Tiefpasses) mit einem Dirac-Stoß (einem unserer Meßwerte) nichts weiter ist, als die Verschiebung dieser Funktion an die Stelle des Dirac-Stoßes und Multiplikation derselben mit dem Betrag des Dirac-Stoßes (Bild 7 links). Und das fällt unserem Atari gar nicht schwer. Wenn wir also für alle Meßwerte, respektive Dirac-Stöße, diese Operation ausführen und alles zusammenaddieren (Bild 7 rechts), haben wir die Spektralfunktionen multipliziert, ohne sie jemals gesehen zu haben, und die Originalfunktion liegt vor. Das einzige Problem ist die Kenntnis der Zeitfunktion des Tiefpaßfilters und die ist - Bild 7 hat es schon verraten - auch wieder eine Spaltfunktion (wir erinnern uns an die Spektralfunktion des Rechteckimpulses). Unter der Maßgabe, daß die Abtastzeit zwischen zwei Werten kurz genug und immer einheitlich lang war und sich die periodischen Spektren auch nicht überlappen, haben wir nun wieder die Originalfunktion wie sie vor der Abtastung war (Bild 1 rechts), und sind glücklich und zufrieden.

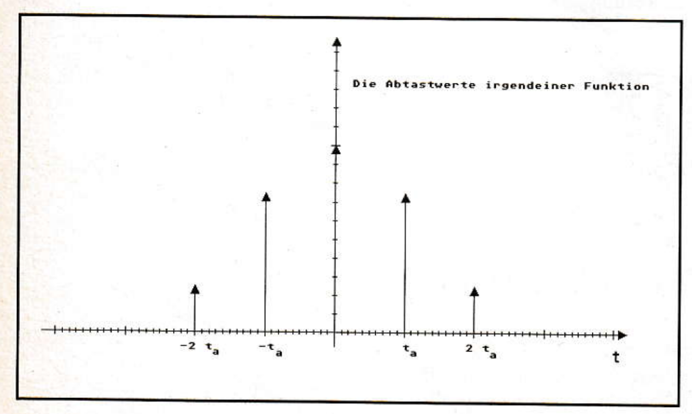
Wenn der CD-Player also zweifach oversamplet, interpoliert er jeweils einen Wert zwischen zweien, die wirklich auf der CD stehen. Oversampling ist demnach nichts weiter als Tiefpaßfilterung, nur daß die Grenzfrequenz des Tiefpasses über dem Hörbereich liegt. Aber wir sagen lieber weiterhin Oversampling, da Tiefpaßfilterung so werbewirksam ist, wie die Feststellung, daß Edelkäse aus verfaulter Milch gemacht wird.

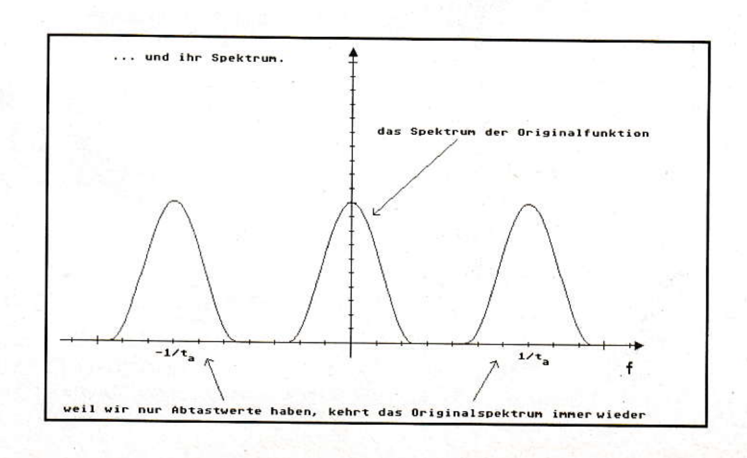
In medias res

Jetzt wollen wir aber endlich auch mal was programmieren. Die Spulen, Kondensatoren und Widerstände lassen wir im Bastlerladen und konstruieren unsere Filter mit RIEMANN2.PRG. Wie vielleicht nicht jeder weiß, ist Riemann nicht nur ein hervorragendes Mathematikprogramm, sondern auch eine richtige Programmiersprache, in der der Liste. Somit steht nun in DX der Abstand zwischen zwei Abtastwerten und in *B* offensichtlich die Anzahl der Meßpunkte. Initial-*Y* ist Null und in der folgenden Zählschleife in *A* von 1 bis *B* wird für jeden Meßpunkt die oben besprochene Verschiebe- und Multiplikationsoperation durchgeführt und aufsummiert. *FIRST(NTHNODE(YLIST,A))* bezeichnet das A-te Element von *YLIST*, also den Betrag des Dirac-Stoßes, und *FIRST (NTHNODE(XLIST,A))* das A-te Element von *XLIST*, also die Stelle, an die wir die Gewichtsfunktion (so heißt nämlich die Zeitfunktion des Tiefpasses) verschieben müssen.

Ergebnis einer jeden Funktion ist immer der zuletzt ausgewertete Ausdruck, also letztlich *Y* in *INTERP*. Der ideale Tiefpaß in der Gewichtsfunktion *G* ist in Kommentarklammern eingeschlossen, da ein Kosinusquadrattiefpaß meist bessere Ergebnisse liefert. Nun aber noch ein Wort zur Spaltfunktion *SI(X)*: Natürlich würde

Bild 4





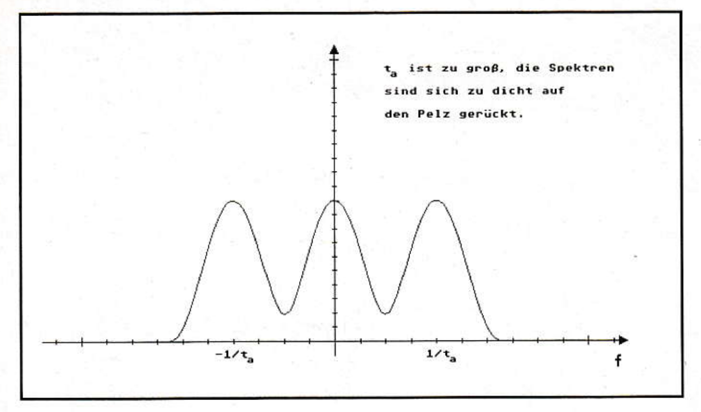


Bild 5

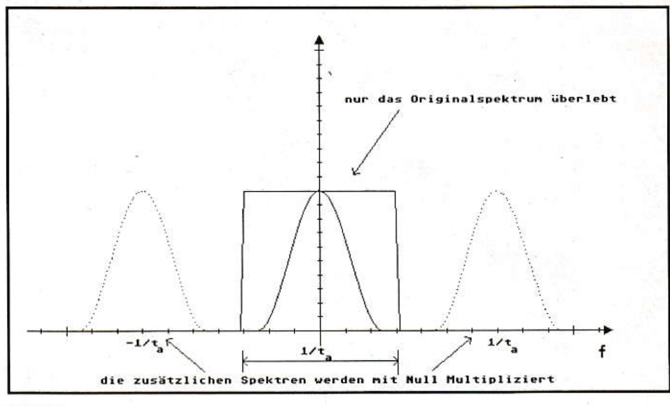


Bild 6

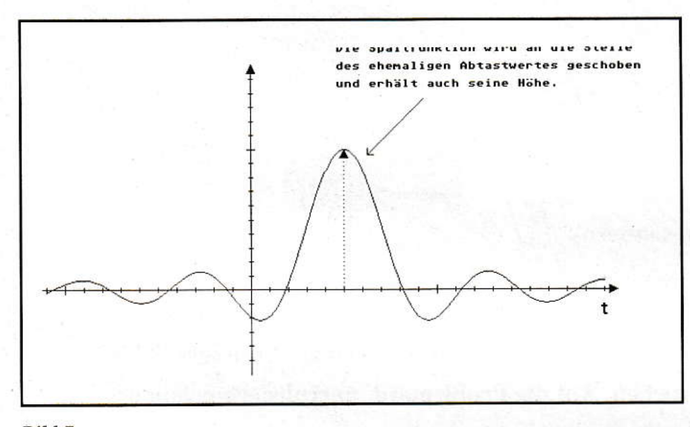
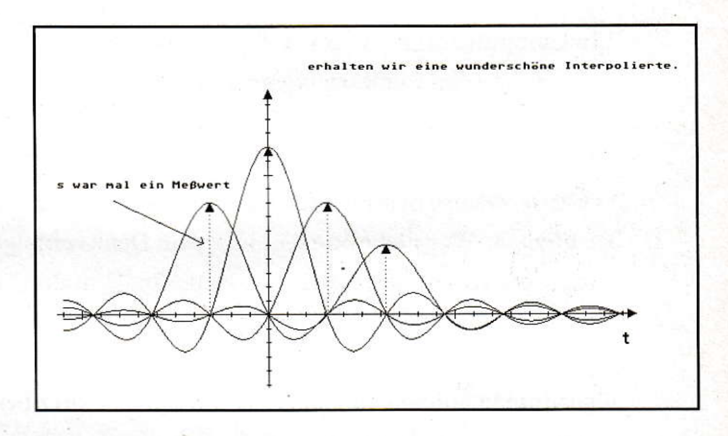


Bild 7



Riemann aufgrund der bereits vorhandenen Intelligenz den Grenzwert *sin(0)/0* auch alleine herausbekommen, aber mit der angegebenen Konstruktion geht es schneller. Warum diese Konstruktion identisch zu ‚if X=0 then 1 else SIN(X)/X' ist, mag der geneigte Leser selbst herausbekommen. Ein kleiner Tip: für die booleschen Operatoren gibt es kein TRUE, sondern nur FALSE oder nicht-FALSE.

Fazit

Auch wenn in diesem Beitrag immer von Zeitbereich und Zeitfunktion die Rede war, kann man statt der Zeit natürlich jede andere Maßeinheit genausogut einsetzen - siehe Fernsehkamera. Digitalfilter in diversen Soundprogrammen, mit denen man die Klangfarbe einstellen kann, arbeiten übrigens nicht nach diesem Verfahren, sie verändern nur die schon vorhandenen Werte mit sogenannten Kammfiltern, welche allerdings ähnlich funktionieren. Wem das alles nicht wissenschaftlich genug war, oder wer einmal andere Tiefpässe oder Bandpässe ausprobieren möchte, dem ist mit [1] sicher weitergeholfen. Und wer [2] besitzt, ist vielleicht gar nicht bis zu dieser Textstelle vorgestoßen. Und nun viel Spaß beim Experimentieren.

Georg Michel

```
% Diese Funktion liefert den interpolierten Wert an der
  Stelle X aus den durch XLIST und YLIST spezifizierten
  Punkten. %

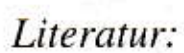
FUNCTION INTERP(X,XLIST,YLIST,%Lokal:%DX,Y,A,B),
  DX:SECOND(XLIST) - FIRST(XLIST),
  B:  LENGTH(XLIST),
  Y:0,
  NLOOP A,1,B,
    Y:Y+FIRST(NTHNODE(YLIST,A))
    *G(X-FIRST(NTHNODE(XLIST,A))),
  ENDNLOOP,
  Y  ENDFUN$

% Das ist die Zeitfunktion des Tiefpasses. Durch Aufhebung
  der Kommentarklammern kann man zwischen einem idealen
  oder einem weicheren cos^2-Tiefpass wählen. %

FUNCTION G(X),
 % SI(##PI*X/DX)                          IDEALER TIEFPASS  %
 SI(2*##PI*X/DX)/(1-(2*X/DX)^2) %         COS^2 - TIEFPASS  %
ENDFUN$

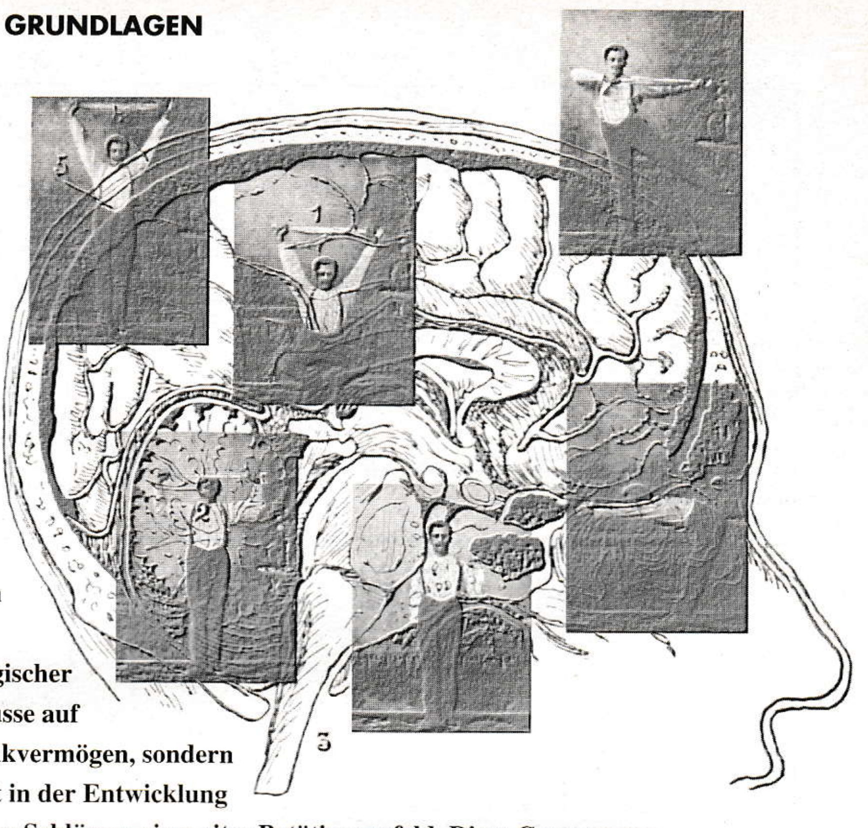
% Die Spaltfunktion. %

FUNCTION SI(X),
  X=0 AND 1 OR SIN(X)/X
ENDFUN$
```

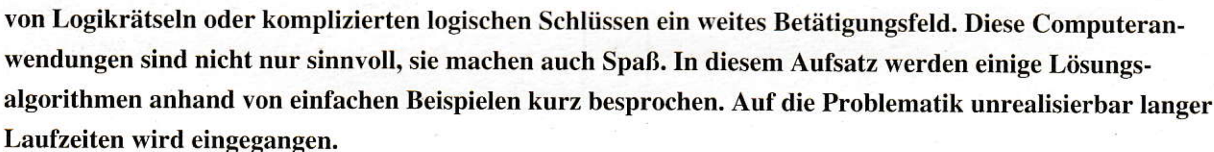

Literatur:

[1] Kreß/Irmer Angewandte Systemtheorie, Verlag Technik 1989
[2] Riemann II-Handbuch

Lösung von Logikproblemen mit dem Computer



Ein Computer eignet sich vorzüglich zur Behandlung von Logikproblemen. Man kann damit logisches Denken trainieren, so beim Lösen von Logikrätseln oder beim Ziehen logischer Schlüsse, ebenso beim Prüfen logischer Schlüsse auf Richtigkeit. Wer nicht nur sein logisches Denkvermögen, sondern auch seine Kreativität fordern möchte, findet in der Entwicklung



von Logikrätseln oder komplizierten logischen Schlüssen ein weites Betätigungsfeld. Diese Computeranwendungen sind nicht nur sinnvoll, sie machen auch Spaß. In diesem Aufsatz werden einige Lösungsalgorithmen anhand von einfachen Beispielen kurz besprochen. Auf die Problematik unrealisierbar langer Laufzeiten wird eingegangen.



Genau genommen: Logikrätsel sind keine Rätsel, denn mit Raten kommt man bei ihnen nicht weit. Auch Wissen ist nicht gefragt. So braucht beispielsweise nicht nach dem Namen eines Flusses in Frankreich mit sechs Buchstaben und einem „n" am Ende gefahndet zu werden. Statt dessen erhält man die Lösung durch Kombinieren und logisches Schließen. Alle dazu nötigen Angaben sind im Logikrätsel enthalten. Um sich von „Rate-Rätseln" deutlich abzuheben, werden Logikrätsel verschiedentlich Logicals, Logeleien oder einfach Denksportaufgaben genannt.

Wie löst man Logikrätsel?

Die Lösung kann mit Bleistift und Papier in Verbindung mit intensiver Kopfarbeit gesucht werden. Man kann sich aber statt dessen eine Stärke des Computers zunutze machen und ihm das Kombinieren und logische Schließen überlassen. Nur: ohne Kopfarbeit geht es trotzdem nicht. Vor der Lösungssuche mit einem Computer müssen die Lösungsprinzipien präzise erarbeitet werden, meist auch viele Details des Computerprogrammes. Das erfordert oft eingehendere Denkarbeit als die Lösungssuche ohne Computer. Aber es stellen sich bald Trainingserfolge ein, die notwendigen Überlegungen fallen auch dem Einsteiger bald leichter. Öftere Aha-Erlebnisse sorgen für Spaß an der Sache. Somit handelt es sich um ein sinnvolles und amüsantes Hobby.

Ein einfaches Beispiel

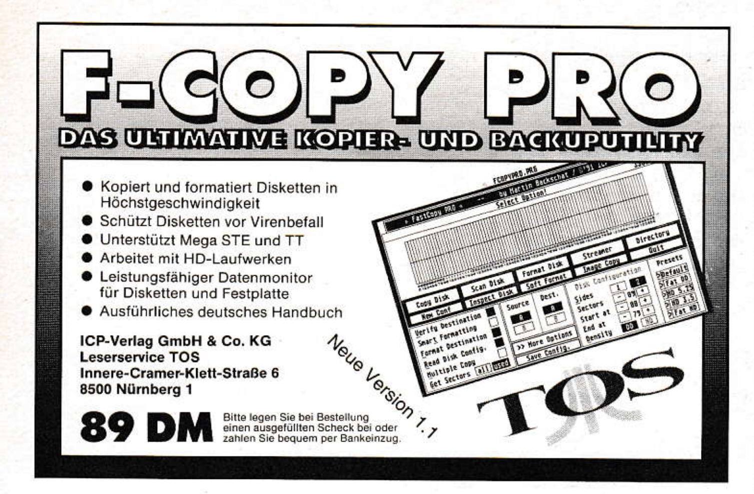
Es gibt viele Arten von Logikrätseln; dies ist ein sehr einfaches Beispiel:

„Wer von Euch hat noch nicht gefrühstückt?" fragte die Mutter ihre vier Kinder nach ihrer unerwartet frühen Rückkehr. Katja, die Älteste, antwortete auf ihre etwas umständliche, aber präzise Art: „(1) Wenn Ingo gefrühstückt hat, dann sind auch Astrid und Marco abgefüttert. (2) Wenn Astrid und ich noch nichts zu uns genommen haben, dann hat es sich Marco schmecken lassen. (3) Wenn dagegen Astrid und ich gefrühstückt haben, dann hat auch Ingo mit uns getafelt. (4) Wenn Marco sein Frühstück intus hat, dann haben entweder Astrid oder Ingo noch nicht gefrühstückt. (5) Wenn Astrid fertig ist und Marco noch nicht angefangen hat, dann hat es mir bereits geschmeckt. (6) Wenn weder Astrid noch Ingo ihr Frühstück angerührt haben, dann bin auch ich noch nicht zum Frühstücken gekommen." Wer hat gefrühstückt?

Eine Denksportaufgabe dieser Art läßt sich lösen, indem per Computerprogramm nacheinander jede der theoretischen Möglichkeiten geprüft wird, ob sie irgendeiner der (hier sechs) Bedingungen widerspricht. Im Falle des Widerspruchs scheidet die betreffende Möglichkeit aus dem Lösungsgang aus. Einzig die allen Bedingungen entsprechende Möglichkeit wird ausgegeben, sie ist die Lösung.

Wieviele theoretische Möglichkeiten gibt es? Bei jedem Kind gibt es zwei Möglichkeiten: hat gefrühstückt oder hat nicht gefrühstückt. Bei den vier Kindern zusammen existieren somit $2 * 2 * 2 * 2 = 16$ Möglichkeiten. Hier sei gleich angemerkt, daß Logikrätsel ohne jegliche Mathematikkenntnisse lösbar sind. Entsprechende Betrachtungen in diesem Aufsatz sind Zusatzinformationen an mathematisch interessierte Leser.

Das genannte Lösungsprinzip kann beispielsweise folgendermaßen realisiert werden: Vier geschachtelte Wiederholungsschleifen erzeugen alle Möglichkeiten. Jede Schleife steht für ein Kind; die Laufvariablen sind mit den Initialen bezeichnet: A für Astrid usw. Jede Laufvariable nimmt nacheinander zwei Werte an: 0 für „hat nicht gefrühstückt", 1 für „hat gefrühstückt". Anschließend wird in sechs Verzweigungen geprüft, ob die Kombination aus den Werten der vier Laufvariablen den Bedingungen 1 bis 6 entspricht oder nicht. So widerspricht eine Kombination beispielsweise der Bedingung 3, wenn nach deren Werten Astrid gefrühstückt hat ($A = 1$) und Katja ebenfalls ($K = 1$), aber



neu Aufdrucke für

T-SHIRTS

-selbst gestalten-selbst drucken-

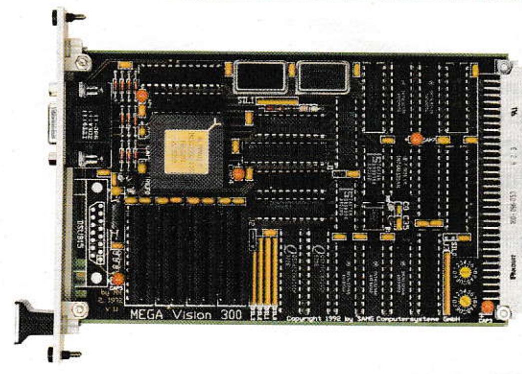
Ohne Spezialfarbband, mit den gängigsten Laserdruckern, Nadeldruckern oder Kopierern auf alle Textilien mit oder ohne Synthetikanteil. Vollwaschbar bis 60 Grad.

Einsteigerpaket nur 39.90 DM

SHIRT-CORNER, Bahnhof Str. 14, 8960 Kempten
Telefon (0831) 10782 ; Fax (0831) 60909
Versandpauschale 6,00 DM Nachnahme oder Vorkasse.

Die Farbvision für Profis:

MEGA VISION 300



- VME-Grafikkarte für Ihren Atari STE/TT
- 1 MByte Bildspeicher in schneller VRAM-Technologie
- flexibler G300-110 MHz Videocontroller
- Farbauflösungen 2, 4, 16, 256 oder 16,7 Millionen
- Bildauflösungen bis zu 1280*1024 programmierbar
- ergonomische Bildwiederholfrequenzen bis 100 Hz
- kompatibel durch optimierten GEM-VDI-Treiber
- qualifizierter Support über Mailbox und Hotline

... Workstationfeeling zum Hobby-Preis: 1.198 DM!

SANG Computersysteme GmbH
Kruppstraße 82 (ETEC)
W-4300 Essen 1
Telefon 02 01/8 20 20-0 Fax:-40



Eine gute Grafikkarte muß nicht teuer sein...

Wollten Sie Ihren Atari Computer schon immer um eine Farbgrafikkarte erweitern, ohne sich mit klapprigen Adapterlösungen zufrieden zu geben ?

Wir bieten speziell für den Einsatz in Atari ST/TT Computern entwickelte Karten...
Und das zu einem erstaunlich günstigen Preis !

Bis zu 256 gleichzeitig darstellbare Farben (aus 16.7 Mio möglichen), Auflösungen bis zu 1712x1224 Punkten und freiprogrammierbare Bildwiederholfrequenzen bis über 120 Hz sorgen für flexible Anpassung an jede Anwendung. Ein vorhandener Sockel ermöglicht das Nachrüsten eines 68881/2 Coprozessors.

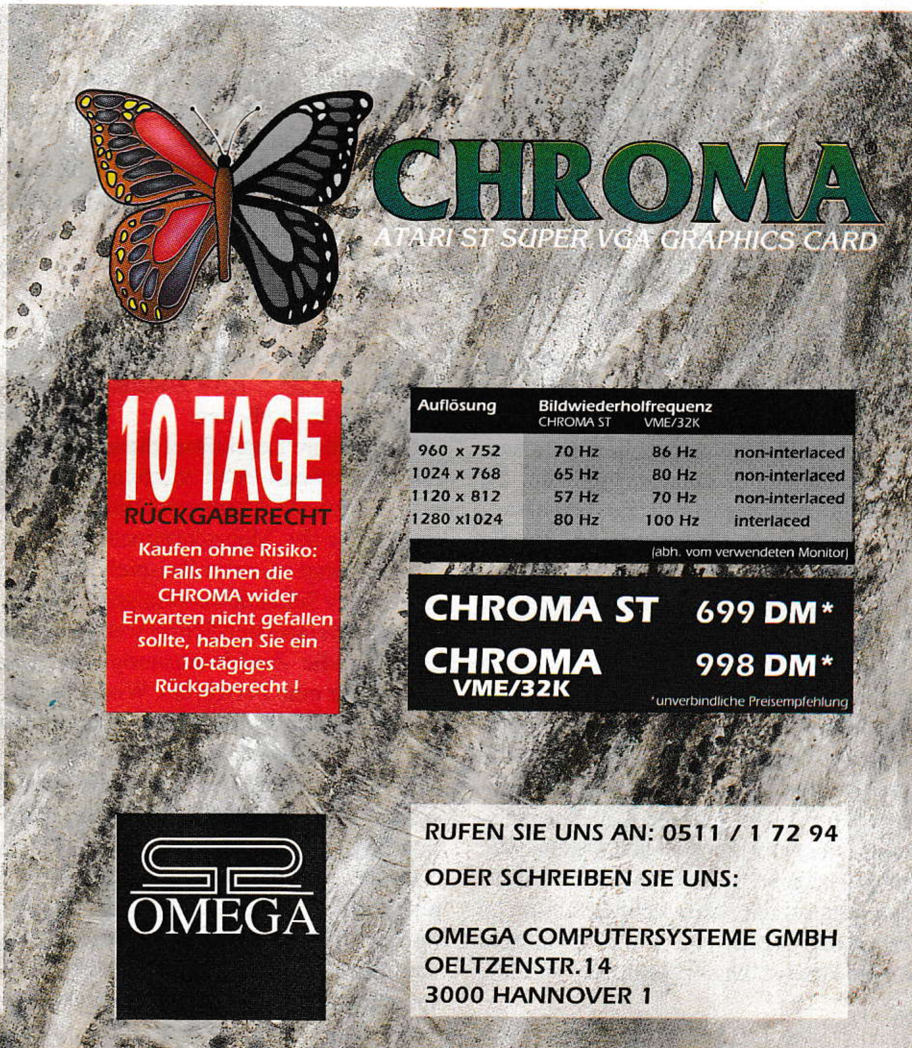
Natürlich arbeiten unsere Karten mit Blitter, diversen Hardwarebeschleunigern und auch mit NVDI zusammen.

Selbstverständlich unterstützt die mitgelieferte Software alle Fähigkeiten der Karte: Hardwarescrolling, Hardwarezoom, 16/256 Farben über VDI Treiber, superschneller VGA Text Modus, getrennte Text/Grafikausgabe auf 2 Monitoren, Videomodusgenerator, GIF-Betrachter.

Die nur 14.3 x 12.6 cm große Platine passt in alle Rechner mit Mega Slot und (per getrennt erhältlichen Adapter) auch in Rechner der 260/520/1040 ST Serien.

Für Rechner mit VME-Bus ist die CHROMA VME/32K erhältlich, die zusätzlich zu den Fähigkeiten der CHROMA die Möglichkeit bietet, bis zu 32768 Farben gleichzeitig darzustellen.
Die erhöhte Pixelfrequenz von 80 Mhz erlaubt noch höhere, flimmerfreie Auflösungen (s. Tabelle).

Besitzer eines MEGA-ST können die CHROMA VME/32K auch heute schon per Adapter in ihrem Rechner einsetzen: Zukunft inklusive !



Auflösung	Bildwiederholfrequenz CHROMA ST	VME/32K	
960 x 752	70 Hz	86 Hz	non-interlaced
1024 x 768	65 Hz	80 Hz	non-interlaced
1120 x 812	57 Hz	70 Hz	non-interlaced
1280 x1024	80 Hz	100 Hz	interlaced

Ingo noch auf sein Frühstück wartet (I = 0). Bei Widerspruch wird der betreffende Programmdurchlauf beendet (z.B. bei BASIC mit einer NEXT-Anweisung, bei Turbo Pascal mit EXIT). Danach wird der Programmlauf mit der nächsten Kombination fortgesetzt. Falls die Kombination allen Bedingungen entspricht, werden ihre Werte mittels der Ausgabeanweisung als Lösung ausgegeben, in unserem Beispiel A = 1, I = 0, K = 0, M = 1, d.h. Astrid und Marco haben gefrühstückt, Ingo und Katja nicht. Ein entsprechendes BASIC-Programm kann z.B. so wie das beigefügte Programm „Frühstück" geschrieben werden.

```
10 REM Programm Frühstück
20 FOR A=0 TO 1 : FOR I=0 TO 1
30 FOR K=0 TO 1 : FOR M=0 TO 1
110 IF I=1 AND NOT(A=1 AND M=1) THEN 200
120 IF A=0 AND K=0 AND NOT(M=1) THEN 200
130 IF A=1 AND K=1 AND NOT(I=1) THEN 200
140 IF M=1 AND NOT(A<>I) THEN 200
150 IF A=1 AND M=0 AND NOT(K=1) THEN 200
160 IF A=0 AND I=0 AND NOT(K=0) THEN 200
190 PRINT A;I;K;M
200 NEXT M : NEXT K : NEXT I : NEXT A
210 END
```

BASIC-Programm „Frühstück"

Otmar	Bankkaufmann	im Großhandel	Diebstahl
Paul	Industriekaufmann	im Bankfach	Betrug
Rolf	Außenh.-Kaufmann	im Gewerbe	Geheimnisverrat
Stefan	Verw.-Ang.	im Außenhandel	Schwänzen
Walter	Steuergehilfe	im Einzelhandel	Unterschlagung

Tabelle: Lösung des Logikrätsels „Allroundmen"

Zu lange Laufzeiten

Der Vorteil des vorgenannten Lösungsprinzips ist seine Einfachheit. Die Lösung größerer Logikrätsel dauert damit aber zu lange. Auch dazu ein Beispiel:

Mit einer Stellenanzeige war ein „Allroundman", nicht älter als 35 Jahre, aber mit 20jähriger Erfahrung im „Kaufmännischen" gesucht worden. Die fünf Bewerber, unter ihnen Paul (wegen des Datenschutzes werden nur die Vornamen genannt), waren offenbar nicht seriöser als die Anzeige. Sie wiesen zwar ihre Ausbildung nach, als Berufserfahrung gaben sie aber nur Tätigkeitsbereiche an. Den Grund ihrer Entlassung von der letzten Beschäftigungsstelle verschwiegen sie ganz, darunter den Verrat von Betriebsgeheimnissen an die Konkurrenz. Jeder hatte einen anderen Beruf erlernt, einen anderen Tätigkeitsbereich angegeben und einen anderen Entlassungsgrund verschwiegen - jeweils welchen? Bekannt ist:

(1) Otmar, der nicht Erfahrungen im Bankfach oder im Außenhandel angegeben hatte, war gelernter Bankkaufmann. (2) Derjenige, der eingehende Erfahrungen im Großhandel angegeben hatte, war wegen Diebstahls entlassen worden. (3) Der Bewerber, der seine letzte Arbeitsstelle wegen Betrugs verloren hatte, hieß nicht Walter, hatte keine Erfahrungen im Außenhandel oder im Gewerbe aufgeführt und war gelernter Industriekaufmann. (4) Rolf hatte Berufserfahrungen im Gewerbe angegeben. (5) Stefan hatte die Papiere wegen notorischen Schwänzens bekommen. (6) Der ausgebildete Steuergehilfe hatte Berufserfahrung im Einzelhandel angegeben. (7) Der gelernte Außenhandelskaufmann war nicht wegen Unterschlagung geschaßt worden. (8) Der Verwaltungsangestellte hatte nicht Erfahrungen im Gewerbe genannt, er war auch nicht wegen Unterschlagung entlassen worden.

Diese Denksportaufgabe handelt von vier Mengen: Der Menge der Personen, der gelernten Berufe, der angegebenen Berufserfahrungen und der Kündigungsgründe. Jede Menge enthält fünf Elemente: die fünf Vornamen, die fünf Berufe usw. Die Lösung besteht darin, die Elemente der verschiedenen Mengen so einander zuzuordnen, daß die (hier acht) Bedingungen eingehalten werden. Man könnte sinngemäß wie beim vorigen Logikrätsel verfahren, indem man dabei alle möglichen Zuordnungen erzeugt und jede Zuordnung (zu je 20 Elementen) auf Widerspruch zu den Bedingungen prüft. Aber: Bereits bei diesem nicht sehr großen Rätsel gibt es 1 728 000 Zuordnungsmöglichkeiten. Sie abzuarbeiten, kann je nach Computer, Programmiersprache und Programm Stunden bis mehrere Tage dauern.

Die Zahl der Zuordnungsmöglichkeiten (Z) steigt mit der Zahl der Mengen (M) und der Zahl der Elemente (E) steil an. Sie kann nach der Formel $Z = (E!)(M-1)$ berechnet werden. Bereits zur Lösung eines derartigen Logikrätsels mit 6 Mengen je 7 Elementen wäre Jahrhunderte erforderlich, um die ca. 3,25 * 1018 Zuordnungsmöglichkeiten abzuarbeiten.

Um größere Logikrätsel dieser Art hinreichend schnell lösen zu können, bedarf es effektiverer Lösungsmethoden, beispielsweise der „Mehrschrittmethode". Mit dieser gelingt es, unser Beispiel in Bruchteilen einer Sekunde bis wenigen Minuten zu lösen. Beim ersten Schritt werden nacheinander die Zuordnungsmöglichkeiten jeweils für eine Person auf Widersprüchlichkeit zu den Bedingungen geprüft und im Falle des Widerspruchs aus dem Lösungsgang ausgeschieden. Hier gibt es 625 solcher Zuordnungsmöglichkeiten, nämlich EM.

Ordnet man die Elemente jeder Menge alphabetisch, wird als erstes die Zuordnungsmöglichkeit Otmar, Außenhandelskaufmann, im Außenhandel, Betrug entsprechend behandelt, dann Otmar, Außenhandelskaufmann, im Außenhandel, Diebstahl usw., zuletzt Walter, Verwaltungsangestellter, im Großhandel, Unterschlagung. Diese drei Zuordnungsmöglichkeiten beispielsweise scheiden aus dem Lösungsgang aus: die erste, weil sie nicht den Bedingungen 1 und 3 entspricht, die zweite wegen Widerspruchs zur Bedingung 1 und die letzte wegen Widerspruchs zu den Bedingungen 2 und 8. Insgesamt werden beim ersten Lösungsschritt 601 der 625 Zuordnungsmöglichkeiten ausgeschieden, ihre Zahl wird somit schnell und rigoros reduziert.

In einem weiteren Lösungsschritt werden jeweils fünf (= E) der verbliebenen 24 Zuordnungsmöglichkeiten miteinander verglichen mit dem Ziel, diejenigen fünf herauszufinden, bei denen alle Elemente (hier 5 * 4 = 20) voneinander verschieden sind. Diese herausgefundenen Zuordnungen sind die Lösung und werden vom Computer ausgegeben, siehe Tabelle „Allroundmen".

Mit falschen Angaben

Bei einer großen Zahl von Logikrätseln sind nicht alle Angaben wahr, beispielsweise:

Carmen:
„Entweder lügt Helga oder Jutta!"
Helga:
„Entweder lügt Carmen oder Jutta!"
Jutta:
„Carmen und Helga sagen die Wahrheit!"

Wessen Aussage ist richtig?
Im Lösungsalgorithmus solcher Denksportaufgaben sind meist neben den Aussagen auch deren Negationen zu berücksichtigen, so auch im beigefügten Pascal-Programm „Luege" zur Lösung dieses Rätsels.

Die drei FOR-Schleifen im Hauptprogramm erzeugen nacheinander die Wahrheitswerte: 0 bedeutet die Annahme fal-

scher Aussage, 1 die Annahme wahrer Aussage. Jede Schleife steht für eine der Personen Carmen (C), Helga (H) und Jutta (J). In der Prozedur *Bedingungen_Ausgabe* wird für jede Person geprüft, ob - bei Annahme wahrer Aussage (z.B. C=1) die anderen Werte zur betreffenden Aussage widersprüchlich sind (z.B. NOT(H<>C)), - trotz Annahme falscher Aussage (z.B. C=0) die anderen Werte der betreffenden Aussage entsprechen (z.B. (H<>C ist).

Wird eines davon festgestellt, dann wird der betreffende Schleifendurchlauf durch EXIT beendet; danach wird die Verarbeitung mit der nächsten Kombination aus der Werten von C, H und J fortgesetzt.

Carmens Aussage wäre wahr, wenn genau eine der beiden anderen Personen lügt, wenn also H und J voneinander verschiedene Werte haben.

Falls der Programmdurchlauf bei keiner der sechs IF-Anweisungen abgebrochen wird, heißt das: Nach den aktuellen Werten von C, H und J haben sich die als wahr angenommenen Aussagen als wahr erwiesen und die als falsch angenommennen Aussagen als falsch. Dies ist die Lösung. Sie wird mit der Ausgabe-Anweisung ausgegeben. In diesem Beispiel besteht die Ausgabe aus drei Nullen, d.h. alle drei Aussagen sind falsch. - Es empfiehlt sich, dieses Ergebnis anhand der Aussagen gedanklich nachzuvollziehen.

```
PROGRAM Luege
VAR
C,H,J:      integer;

PROCEDURE Bedingungen_Ausgabe
BEGIN
 IF (C=1) AND NOT (H<>J) THEN EXIT;
 IF (C=0) AND (H<>J) THEN EXIT;
 IF (H=1) AND NOT (C<>J) THEN EXIT;
 IF (H=0) AND (C<>J) THEN EXIT;
 IF (J=1) AND NOT ((C=1) AND (H=1)) THEN EXIT;
 IF (J=0) AND ((C=1) AND (H=1)) THEN EXIT;
 writeln(C,H,J)
END;  (*der Prozedur*)

BEGIN  (*Hauptprogramm*)
 FOR C:=0 TO 1 DO
  FOR H:=0 TO 1 DO
   FOR J:=0 TO 1 DO
    Bedingungen_Ausgabe;
END  (*Hauptprogramm*)
```

Pascal-Programm „Luege“

Mit indirekten Angaben

In sogenannten Metarätseln, die auch als „Rätsel über Rätsel“ bezeichnet worden sind, fehlt mindestens eine zur Lösung erforderliche Angabe. Dafür ist angegeben, ob jemand, der im Besitz zusätzlicher Informationen ist, die Denksportaufgabe lösen könnte oder nicht. Mit dieser Zusatzinformation über die Lösbarkeit kann nun das Rätsel gelöst werden. Dies ist ein Beispiel dieser Art:

Auf die Frage des Deutschlehrers, ob sie und ihre Freundinnen Elke und Gabi bereits einen Ausbildungsplatz gefunden hätten, antwortete Birgit: „Wenn Elkes Bewerbung erfolgreich war, dann hat Gabi eine Ausbildungsstelle oder ich keine. Wenn ich einen Ausbildungsplatz habe, dann ist Gabi noch auf der Suche. Und ob ich eine Ausbildungsstelle habe oder nicht, das habe ich Ihnen vorhin schon gesagt.“ - „Richtig“, erinnerte sich der Pädagoge; nach kurzem Nachdenken tat er kund, daß er jetzt auch über die beiden anderen Bescheid wisse. Wer von den dreien hat einen Ausbildungsplatz?

Man kann dieses Logikrätsel wie das Beispiel „Frühstück“ lösen, wobei die Bedingungen nur aus Birgits wörtlicher Rede abgeleitet werden. Allerdings gibt der Computer fünf Ergebnisse aus: nach vier von ihnen hat Birgit noch keinen Ausbildungsplatz, nach einem Ergebnis hat sie eine Stelle. Wäre dem Lehrer vorher mitgeteilt worden, Birgit habe keine Ausbildungsstelle, dann hätte er nicht dahinter kommen können, wie es um Elke und Gabi steht: jede von ihnen hätte einen Ausbildungsplatz haben können oder nich . Also muß ihm Birgit gesagt haben, daß sie bereits eine Stelle habe. Die entsprechende Ausgabe des Computers zeigt, daß die beiden anderen noch keine Ausbildungsstelle haben.

Selbst Logikrätsel erstellen

Im Prinzip wird man ein Programm zur Lösung eines Logikrätsels weitgehend auch zur Erstellung eines gleichartigen Rätsels verwenden können. In jedem Falle aber müssen die Anweisungen, die von den speziellen Bedingungen des einen Rätsels abgeleitet sind, durch entsprechende Anweisungen des zu erstellenden Logikrätsels ersetzt werden. Jedes selbst erstellte Logikrätsel ist sorgfältig darauf zu prüfen, daß es keine inneren Widersprüche enthält und in bezug auf die Lösung eindeutig ist. Auch soll es möglichst keine redundanten, also keine überflüssigen Bedingungen enthalten. Ein Computer ist zur Entwicklung und zur Prüfung des Bedingungssystems fast unentbehrlich.

Logische Schlüsse

Sehr einfach ist es, z.B. aus den Prämissen „Alle Boxweltmeister sind Sportler“ und „Max ist Boxweltmeister“ den Schlußsatz zu bilden, also den logischen Schluß zu ziehen: „Max ist Sportler“. Dazu benötigt man keinen Computer. Ist aber ein Schlußsatz aus mehr als zwei Prämissen abzuleiten, ist ein Computer meistens hilfreich. Mit einem Computer kann man aber auch prüfen, ob der Schluß richtig gezogen worden ist, so z.B., ob der aus den Prämissen „Wenn ich lese, dann brauche ich Licht“ und „Ich brauche Licht“ gebildete Schlußsatz „Ich lese“ richtig ist. Möglich ist folgendes Vorgehen: Zunächst werden mit dem Computer alle Zuordnungsmöglichkeiten ausgeschieden, die nicht den Prämissen entsprechen. Danach wird jede der verbliebenen Zuordnungsmöglichkeiten geprüft, ob sie dem Schlußsatz widerspricht. Wenn dabei der Computer einen Widerspruch feststellt, gibt er eine entsprechende Mitteilung aus (z.B. „Schluß falsch“) - wie er es auch bei diesem sehr einfachen Beispiel täte.

Voraussetzungen

Spezielle Software ist zur Behandlung von Logikproblemen nicht erforderlich. Die nötigen Programme kann man selbst erstellen. Auch an die Hardware werden keine besonderen Anforderungen gestellt. Ein Einsteiger hat keine allzu großen Schwierigkeiten zu erwarten: Wie die Beispiele zeigen, gibt es (viele) Denksportaufgaben, die mit einigen nicht allzu schwierigen Überlegungen und nur geringen Programmierkenntnissen lösbar sind. Trotzdem wird es nicht langweilig - selbst auf lange Sicht nicht. Bei den vielen verschiedenartigen Logikrätseln unterschiedlichen Schwierigkeitsgrades müssen sich auch „Fortgeschrittene“ immer wieder etwas einfallen lassen, sei es zur Lösungsmethode selbst oder zu Details des Computerprogrammes - was ja der Sinn der Sache ist.

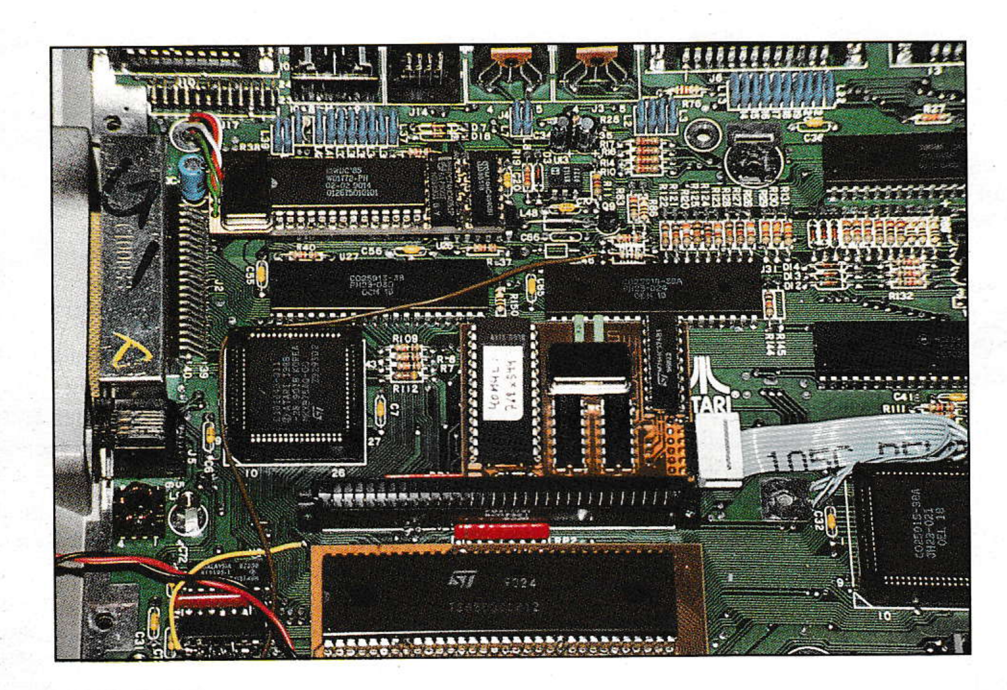
Günter Süßmuth

Literatur:

Logiktraining mit Logikrätseln - Unterhaltsame Rätsel lösen mit Hilfe von BASIC und Turbo Pascal, Vaterstetten 1990

Logikrätsel und logische Schlüsse - Anspruchsvolle Logikrätsel lösen und logische Schlüsse ziehen mit Hilfe von BASIC und Turbo Pascal, Vaterstetten 1991

Und es geht doch!



In dieser Artikelserie wird beschrieben, wie man mit sehr wenig finanziellem Aufwand einen ATARI ST gleich in mehrfacher Hinsicht aufbohrt. Eine DEUTLICHE Warnung jedoch vorweg: Ohne ein gerüttelt Maß an Erfahrung und/oder Selbstvertrauen im Basteln an elektronischem Gerät sollten Sie UNBEDINGT die Finger von diesem Projekt lassen!

ATARI (MEGA) ST mit mehr als 8MHz!

Oft wurde dieses Thema diskutiert, aber niemand wollte es so recht für möglich halten. Hier ist der Beweis: der ATARI- ST ist nach einigen Modifikationen durchaus in der Lage mit höheren Taktfrequenzen als 8MHz zu laufen. Im einzelnen wird nach dem Umbau in Ihrem ATARI folgendes anders sein als vorher:

- Die Maschine läuft nicht mehr mit 8 MHz, sondern mit 10 oder ggf. mehr und ist auch bei allen Operationen außer Disk-Zugriff entsprechend schneller, unabhängig vom Wetter und eventuellen Cache-Trefferquoten.
- Es ist alles vorbereitet, um ein bis zwei HD-Laufwerke anzuschließen, Betrieb mit normalen Laufwerken und gemischt ist aber natürlich auch möglich.
- Eine sehr einfache, aber wirkungsvolle Bildschirmvergrößerung fällt ebenfalls 'als Abfallprodukt' an.
- Sie müssen sich, und das ist ganz klar ein Wermutstropfen, für entweder Farb- oder Monochrombetrieb bei gleichzeitiger Bildschirmvergrößerung oder für alternativen Farb- und Monobetrieb OHNE Bildschirmvergrößerung entscheiden. Dabei gleich die Warnung, daß vom Autor der Farbbetrieb NICHT getestet wurde, aber kein prinzipielles Problem dabei besteht!
- Manch eine andere Hardware-Erweiterung funktioniert danach möglicherweise nicht mehr. Nicht betroffen davon sind nachgeprüftermaßen AT-Speed C16 sowie die RESOLUTION von GengTEC, denn diese arbeiten im Rechner des Autors (ehemaliger 260ST Baujahr uralt), sowie die SCSI-Adapter von ICD. Der Adapter für die c't-Billiglösung läuft nur sicher, wenn er komplett mit ALS- statt LS-Chips bestückt wird (der eine vorhandene HCT-Chip ist schnell genug).

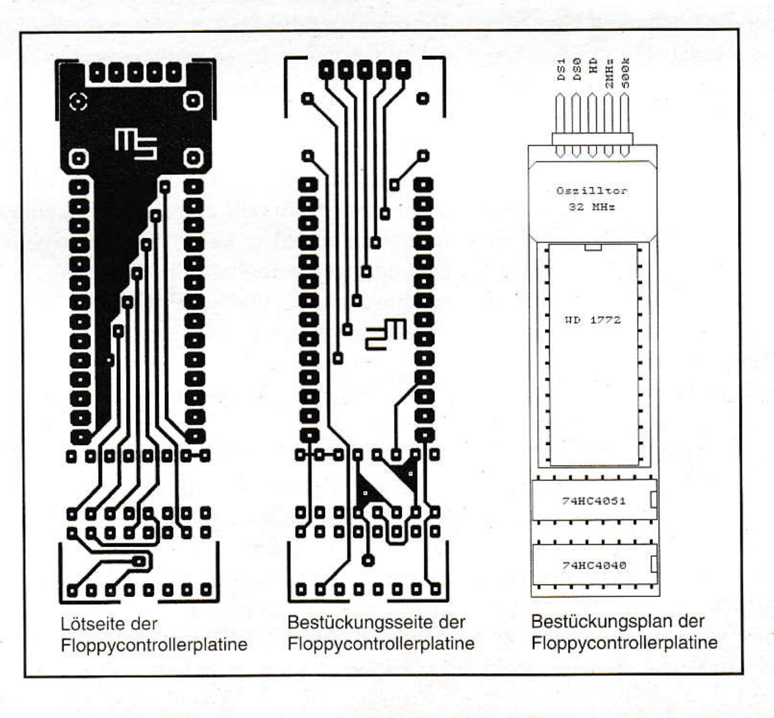


Bild 1: Floppy-controller-Adapterplatine

GAL-Programmiergerät MGP 16/20

Entwicklungssystem für Logikschaltungen

Leistungsstarkes Programmiergerät für die Realisierung logischer Schaltungen (NOR-, NAND-, NOT-,... Gatter) mit den gängigen GAL-Typen 16v8 und 20v8 und deren A-Typen. Das Gerät wird an die Druckerschnittstelle (parallel - Centronics) angeschlossen. Die menügesteuerte Software ermöglicht ein bequemes und sicheres Arbeiten. Integrierter 2-Pass-Logic-Compiler, der logische Gleichungen in JEDEC-Dateien übersetzt. Optimierung der Gleichung nach Quine-McCluskey.

Bestellnr.: 890900, Fertiggerät, DM 229,-*
Bestellnr.: 890901, Platine, Software, DM129,-*

MSA

SCSI-Adapter

Schneller SCSI-Adapter zum Anschluß von SCSI-Geräten an den Atari ST. Hohe Übertragungsraten, macht das angeschlossene SCSI-Gerät uneingeschränkt bootfähig, kompatibel zu den meisten erhältlichen SCSI-Festplatten (z.B. Seagate ST157N, Quantum Pro80, Syquest SQ555 usw.), unterstützt alle SCSI-Kommandogruppen, gepufferter DMA-Bus, Abschluß max. 4 SCSI-Geräten, Hardware-Schreibschutz, inkl. Festplattentreiber.





Bestellnr.: 900810, Fertiggerät, DM 259,-*
Bestellnr.: 900811, Platine, GALs, Software, DM 149,-*

Junior Prommer

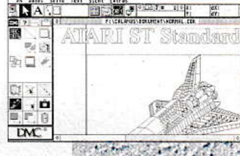
EPROM-Programmiergerät

Programmiert alle gängigen EPROM-Typen und deren CMOS-Typen (2716-27011). Komfortable Software mit Zerlegung in High- und Low-Byte, 5 Programmieralgorithmen, Hex-/ASCII-Monitor mit vielen Edierfunktionen. Leichtes Erstellen von EPROM-Bänken durch Software-Unterstützung. Mit optionalem Adaptersockel Mega-Modul lassen sich auch 32pol. EPROMs (27010-27080) brennen.Das Gerät wird an die Druckerschnittstelle (parallel - Centronics) angeschlossen.

Bestellnr.: 880310, Fertiggerät, DM 229,-*
Bestellnr.: 88031, Platine, Software, DM 59,-*
Bestellnr.: 880312, Leergehäuse, DM 39,90*
Bestellnr.: 880313; Zusatzadapter Mega Modul, DM 99,-*

PixelWonder

Eine wirklich scharfe Sache!







PixelWonder erhöht die Bildschirmauflösung eines 260, 520, 1040 ST und Mega ST. Eigener Videoprozessor sorgt für maximale Leistung. PixelWonder ist frei konfigurierbar und unterstützt SM124 (je nach Baureihe bis zu 768*528 Pixel) **und** Multi-Sync-Monitore (z.B. 832*624 Pixel). Bildwiederholfrequenz bis zu 94Hz. PixelWonder benutzt das original Atari-Betriebssystem. Alle auflösungsunabhängigen Programme laufen. PixelWonder ist abschaltbar, daher auch zu Spielen voll kompatibel. Leider nicht lauffähig mit 1040 STE und MegaSTE. Der Einbau erfolgt durch Auflöten auf den Prozessor und Anlöten 5 zusätzlicher Leitungen. Löterfahrung ist erforderlich!
AutoSwitch: Da einige Programme fest auf Standardauflösungen ausgelegt sind, schaltet PixelWonder beim Start dieser Programme automatisch auf die normale Auflösung zurück.
DoubleScan-Modus: Für Farbdarstellungen bietet PixelWonder ein Zeilenverdopplungsverfahren, das eine deutlich verbesserte Bildqualität bewirkt.

Pixel Wonder Grafikwunder

Die fantastische Grafikerweiterung für ATARI ST

Bestell-Nr.: 910400, DM 148,-*

*Alle Preise sind unverbindliche Preisempfehlungen. Bei Nachnahmebestellung wird eine NN-Gebühr von DM 8,- fällig. Auslandsbestellungen **nur** gegen Vorauskasse

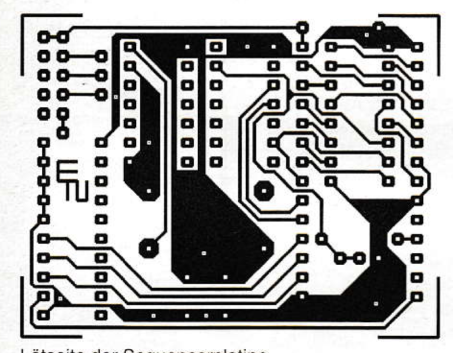
MAXON Computer GmbH
Schwalbacher Straße 52
W-6236 Eschborn
Tel. 06196/481811
Fax 06196/41885



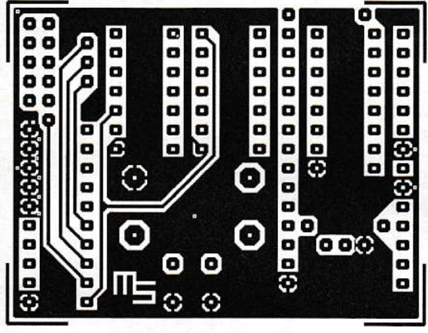
Fleißarbeit

Wie das oft so ist, was man an Geld spart, investiert man an Zeit. Zur Durchführung dieses Umbaus sind eine Reihe von Chips auszulöten und zu sockeln bzw. gegen andere auszutauschen. Doch zunächst sollten Sie unter Verlust der Garantie Ihren Rechner aufschrauben (vorher den Netzstecker ziehen und 5 min. warten!) und einige Voraussetzungen überprüfen:

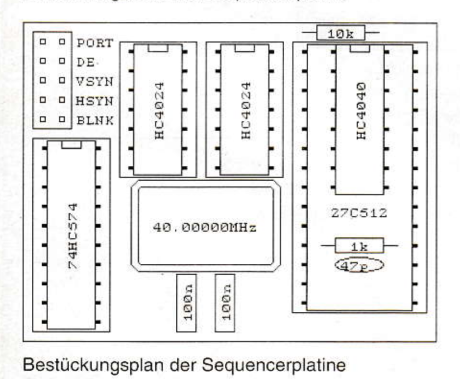
- Besitzt der ATARI einen Quarz (im Shifter-Blechgehäuse, bei 260/520) oder einen Quarzoszillator (hinter dem MEGABUS)? Der Umbau der Modelle mit Quarz ist komplizierter und wird nicht so detailliert beschrieben werden. Insbesondere ist bisher kein 1040 einem solchen Umbau unterzogen worden. Obwohl er auch dort wohl funktioniert, ist der geneigte Leser hier im Detail auf reichlich Eigeninitiative angewiesen. Am einfachsten ist das Tuning in MEGA STs zu bewerkstelligen.



Lötseite der Sequencerplatine



Bestückungsseite der Sequencerplatine



Bestückungsplan der Sequencerplatine

Bild 2: EPROM-Sequencer-Platine

- Die RAMs haben eine Zugriffszeit von 120ns oder weniger? Einem Betrieb mit 10MHz steht nichts im Wege! Für 12MHz-Betrieb brauchen Sie 80ns oder schneller!
- Sie haben bei den TOS-EPROMs, so Sie welche haben, nicht gespart und 250ns-Typen genommen? 150ns sollten sie haben, um problemlos zu funktionieren (leider steuert ATARI die ROMs nicht über Output Enable, wobei sie per se schneller sind, sondern über Chip-Select, was recht gemächlich ist und letztendlich der aufgedruckten Geschwindigkeit entspricht). Original TOS-ROMs sind, soweit dem Autor bekannt, keinesfalls zu langsam.
- Der Floppy-Controller WD 1772 trägt den Zusatz -02 in der Typenbezeichnung? Dann wird HD-Betrieb tatsächlich möglich sein; bei anderen Controllern wird er wahrscheinlich nicht funktionieren. Für Ihr normales Laufwerk können Sie ihn aber weiterbenutzen.

Wie funktioniert das nun eigentlich?

Jahrelang hieß es, wegen seines in den CPU-Zyklus verwobenen Video-Zugriffs sei der ATARI nicht mit mehr als den serienmäßigen 8MHz zu betreiben. Dabei ging man immer davon aus, einfach nur einen anderen Taktgeber einzubauen und dann ginge es (schneller). So einfach funktioniert das nicht, doch werfen wir einen Blick auf alle dabei auftretenden potentiellen Knackpunkte:

- Bei höheren Taktfrequenzen wird das gesamte Bildschirm-Timing auch schneller werden, da es im ATARI aus dem Systemtakt abgezählt wird. Leute, die einen leistungsfähigen Multiscan an ihrem ATARI betreiben, werden darüber nur milde lächeln, doch standardmäßige Monitoren darf man nicht mehr anschließen, es sei denn, man baute eine kleine Hardware, die ein Video-Timing wie bisher erzeugen und es dem ATARI unterschieben würde. Eine solche Hardware in einfachster und billigster Form wird im Laufe dieses Artikels beschrieben werden.
- Höhere Taktfrequenzen könnten Custom-Chips, RAMs und ROMs in Zeitnot bringen. Das tritt auch ein, doch lassen sich im RAM-Timing bei Treibern und Latches wertvolle Nanosekunden sparen. ROMs und RAMs kann man ggf. gegen schnellere austauschen. Die Custom-Chips vertragen nach Erfahrung des Autors den höheren Takt klaglos. Unrühmliche Ausnahme ist (mal wieder) der Blitter, der zumindest in einem MEGA 2.5 des Autors bei 12MHz NICHT mitspielt, wohl aber bei 10MHz.
- Die Töne, die der Soundchip erzeugt, stimmen nicht mehr. Daher werden wir ihn wie bisher mit 2MHz versorgen müssen.
- Die Tastaturschnittstelle mit ihrer seriellen Übertragung hängt auch am 8MHz-Systemtakt, weswegen wir auch hier den Originaltakt anlegen müssen. Als angenehmer 'Nebeneffekt' bleibt dabei die Funktion von MIDI erhalten ...

Monitorsteuersignale

Zum Betrieb seines SM-124-Monitors erzeugt der ST die üblichen Synchronsignale VSYNC und HSYNC. Ersteres bestimmt die Bildwiederholfrequenz und tritt 71.3mal und damit häufiger als 70mal pro Sekunde auf, weswegen das Bild so schön flimmerfrei ist. Letzteres liegt mit einer Frequenz von 35.7kHz vor, was für Fernseher und preiswerte TTL-Monitore deutlich zuviel ist. Sogenannte VGA-Monitore verkraften diese Zeilenfrequenz jedoch durchaus teilweise, und Multiscans allemal. Die Bildqualität dieser Monitore am ST ist jedoch recht mäßig, weil der SM 124 sich das katastrophale Videosignal (=Träger der Bildinformation) des ST intern regeneriert, wogegen alle ebengenannten Fremdmonitore die flachen Flanken getreulich und leider entsprechend unscharf wiedergeben. Daher ist es durchaus sinnvoll, den guten alten SM 124 auch mit der erhöhten Auflösung weiterzubenutzen. Dabei muß aber beim Erstellen des Sequencer-EPROMs peinlichst darauf geachtet werden, das abgebildete (Bild 7) Original-Timing des SM 124 einzuhalten! Dieses Timing entstammt übrigens den Original-Serviceunterlagen.

Die Leitung 'DE', von der im Artikel die Rede ist, teilt dem Shifter mit, wann und für wie lange er Words aus dem Hauptspeicher als Pixel ausgeben soll. 'BLNK' schließlich tastet den Bildschirm dunkel, ist jedoch nur für die Farbmodi relevant und wird vom abgedruckten EPROM-Generator einfach synchron zu 'DE' gesetzt.

Werfen Sie einen genauen Blick auf die untenstehenden Zeitdiagramme einer Zeile resp. eines Bildes, bis Sie die Vorgänge richtig verstanden haben, bevor Sie den Inhalt des Sequencer-EPROMs maniupulieren!

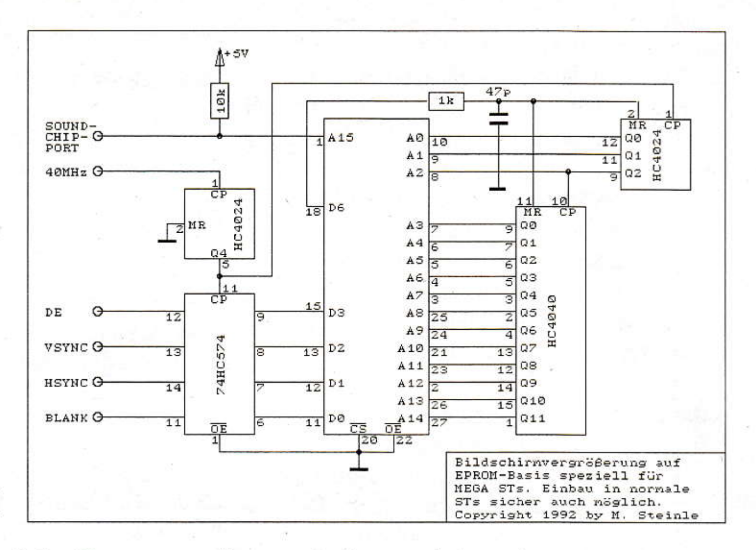


Bild 3: Schaltung EPROM-Sequencer

- Last not least muß der Floppycontroller mit weiterhin 8MHz versorgt werden, sonst kreieren Sie sich ab sofort ein höchst persönliches Diskettenformat, was Sie sicher nicht wollen. Bei der Gelegenheit bietet sich an, gleich HD-Betrieb vorzusehen, weshalb in dieser Umbauanleitung auch die 793. Anleitung für einen HD-Adapter zu finden sein wird.

In medias res ...

... geht es nun nach soviel Vorgeplänkel. Zum Auslöten aller im folgenden aufgeführten Chips sollten Sie sich, in der Reihenfolge der Tauglichkeit, irgendwelche der folgenden Utensilien zurechtlegen:

- Entlötkolben mit Vakuumpumpe (geht wunderbar, bloß hat nicht jeder so ein Ding)
- Entlötpumpe UND Entlötlitze in Verbindung mit einem nicht zu schwächlichen Lötkolben, besonders wegen letzterer
- Von Kanülen (siehe diverse Quicktips) und ähnlichem wird dringend abgeraten, doch möge jeder nehmen, womit er am besten zurechtkommt.

Vier Tips noch dazu:

- Auch sorgfältig von Zinn befreite Pins neigen dazu, noch an einer Ecke zu 'kleben'. Daher bewegt der Autor jeden einzelnen Pin mit einer Flachzange, bevor er am ganzen Chip hebelt. Dieses Verfahren ist sehr platinenschonend, wenn vorher gut abgesaugt wird.
- Der zweite Tip betrifft den Lötkolben: Entgegen landläufiger Meinung ist der Autor der Auffassung, daß es insbesondere beim Entlöten keine zu starken Lötkolben gibt. Immerhin müssen an einigen Stellen große Kupferflächen mit erhitzt werden, um das Zinn absaugen zu können.
- Das tödlichste für Platinen ist seitliches Ausüben von Druck auf Löt-Pads, weil die Verklebung des Kupfers mit dem Basismaterial unter Hitzeeinwirkung das schwächste Glied in der Kette ist. IMMER OHNE DRUCK nach der Seite die Pins erhitzen!
- Nicht vollständig freigesaugte Pins vor dem erneuten Absaugen frisch löten, da so der Wärmeübergang vom Lötkolben weit besser ist!

Auszulöten aus der Platine sind folgende Bauteile:

- CPU 68000-8, leider. Diesen Chip funktionsfähig unter Zurücklassung einer intakten Platine auszulöten ist zweifellos das Entlöter-Meisterstück.
- Floppycontroller WD 1772. Mit seinen nur 28 Pins ist das gegenüber der CPU direkt ein Kinderspiel.
- die vier Treiber resp. Latches in der Nähe der MMU (2x74LS244 und 2x74LS373)
- der integrierte Quarzoszillator in der Nähe des Megabusses. In Modellen mit normalem Quarz bleibt dieser erstmal an Ort und Stelle.

Die CPU und die Treiber/Latches werden übrigens nicht weiterverwendet. Sie könnten sie also auch abzwicken vor dem Auslöten, doch sooo gut wie oft behauptet funktioniert das auch nicht, und besonders schonend für die Leiterplatte ist es wegen der großen Kräfte nicht gerade.

... geht er noch?

Wenn alle Teile ohne Schaden für die Leiterplatte draußen sind, sollten Sie überall Sockel (scheuen Sie sich nicht, die paar Pfennige für gedrehte Kontakte auszugeben ...) einlöten (für den Quarzoszillator entweder vier Einzel-Pins verwenden oder von einer 14poligen Fassung die Pins 2-6 und 9-13 abkneifen) und die neuen Bauteile wie folgt einstecken:

* Betriebssystem : TOS 2.05 vom 05.12.1990
* Referenzsystem : TOS 1.04
* CPU : M68000
* Blitter : abgeschaltet
* Taktfrequenz : 12MHz

	normal	NVDI
Textausgabe	152%	1222%
Linien	152%	450%
Rechtecke	151%	621%
Polygone	153%	304%
Kreise/Ellipsen	154%	555%
Rasteroperationen	151%	448%
Attributfunktionen	151%	734%
Auskunftsfunktionen	151%	547%
ESCAPES	151%	395%
BIOS-Ausgabe	147%	302%
GEMDOS-Ausgabe	148%	949%
AES-Objekt-Ausgabe	159%	505%

* Betriebssystem : TOS 2.05 vom 05.12.1990
* Referenzsystem : TOS 1.04
* CPU : M68000
* Blitter : siehe unten
* Taktfrequenz : 10MHz

	normal	Blitter	NVDI	Blitter+NVDI
Textausgabe	126%	228 %	1015%	1573%
Linien	126%	241 %	373%	517%
Rechtecke	125%	654 %	514%	994%
Polygone	127%	142 %	252%	263%
Kreise/Ellipsen	127%	130 %	461%	461%
Rasteroperationen	125%	911 %	364%	1014%
Attributfunktionen	125%	125 %	607%	608%
Auskunftsfunktionen	125%	125 %	453%	453%
ESCAPES	125%	156 %	329%	329%
BIOS-Ausgabe	122%	144 %	252%	252%
GEMDOS-Ausgabe	123%	137 %	791%	791%
AES-Objekt-Ausgabe	131%	176 %	419%	465%

- ein neuer 10-12MHz-68000, je nach Optimismus bezüglich der erzielbaren Taktfrequenz
- zwei neue 74ALS244 oder 74F244 statt der 74LS244
- zwei neue 74ALS373 oder 74F373 statt der 74LS373
- den alten Floppycontroller

Weiterhin muß die Terminierung einiger Leitungen verbessert werden: Zunächst der Adreßbus (Pin 29-48/50-52 der CPU), im Original 'terminiert' mit 10k-Netzwerken (bei 260/520/1040ST) oder immerhin 4k7 (im MEGA ST), muß mit im Ergebnis ca. 3k3 abgeschlossen werden. Das geht, indem man zu den vorhandenen 10k-Netzwerken welche mit 4k7 parallelschaltet (bzw. umgekehrt), oder man macht sich die Arbeit und lötet neue 3k3-Netzwerke ein. Noch wichtiger ist der Ersatz des 1k-Pull-Up-Widerstands an der Reset-Leitung des Prozessors (von Pin 18 aus mit dem Ohmmeter suchen!) durch 560 Ohm (beim MEGA 1 ist es R4 unter dem Netzteil, beim 260ST ist es einer der 1k-Widerstände in der Nähe der CPU, zumeist R31). Unterläßt man letzteres, so bootet der Rechner mit 12MHz nicht mehr, weil die CPU als zweiten Befehl einen Peripherie-Reset durchführt und die Peripherie (namentlich der GLUE und mit ihm das ROM) wegen des lansamen RESET-Signals zu spät aus dem Tiefschlaf zurückkommt (das dürfte auch der Grund sein, warum manche Rechner mit Turboboards dann nicht mehr booten ...). All diese Werte bewegen sich übrigens weit innerhalb der CPU-Spezifikation! Bei vielen Rechnern ist es auch erforderlich, das Videosignal im Pegel zu erhöhen (Symptom: an den Wortgrenzen auf dem Bildschirm leichte Störungen). Das geht, indem man vom Shifter-Pin 30 ausgehend mit dem Ohmmeter einen 1k-Widerstand (MEGA 1: R82, 260ST: R45, trotzdem kontrollieren!!) sucht, der daran hängt, und diesen gegen 560 Ohm austauscht. Dabei wird die Grundhelligkeit des Bildes allerdings höher, was jedoch mit den Kontrast- und Helligkeitsstellern am Monitor in den Griff zu bekommen ist.

Wenn Sie jetzt den Rechner probehalber in Betrieb nehmen (bitte Vorsicht mit der offenen Netzspannung im Netzteil), sollte er GANZ NORMAL funktionieren, denn schließlich haben wir bisher noch nichts geändert, sondern nur Arbeit und Geld investiert.

Als ersten Schritt hin zum aufgebohrten ATARI werden wir nun den Floppycontroller, den Soundchip und die ACIAs mit eigenen Taktleitungen versehen. Dazu sollten Sie die erste kleine Leiterplatte (Bild 1) folgendermaßen bestücken:

- die Löcher für die Pins des Floppycontrollers müssen so groß gebohrt sein, daß man Pins von gedrehten Sockeln bis zum Kragen durchschieben kann. Die Pins von oben und unten vorsichtig verlöten, ohne die Kelche zuzulöten und ohne die eigentlichen Pins mit Zinn zu benetzen. Dabei hat es sich bewährt, die Pins in einer 28poligen Fassung zu führen, damit sie hinterher schön in Reih' und Glied stehen. Pin 18 (das ist der Clock-Pin) schließlich so weit abfeilen, daß er, im ATARI eingebaut, ganz sicher nicht mit dem darunterliegenden Sockel Kontakt bekommt!
- alle Durchkontaktierungen (sieben an der Zahl unter dem WD 1772 und eine unter dem 74HC4040) herstellen
- den 74HC4040 und den 74HC4051 ohne Sockel bestücken und beidseitig verlöten. Für diese Bausteine ist HCT vorteilhaft, jedoch schwerer erhältlich
- den 32MHz-Oszillator von der Hauptplatine bzw. im Falle eines 260/520/1040ST einen zusätzlich gekauften, an die vorgesehene Stelle einlöten, ebenfalls ohne Sockel
- oberhalb des Oszillators fünf Kabel von vorläufig ca. 30cm Länge einlöten
- die bestückte Leiterplatte sorgfältigst sichtkontrollieren, danach den WD 1772 einstecken und die gesamte Anordnung in den Floppycontroller-Sockel auf der Hauptplatine stecken
- Nur beim Umbau eines MEGA ST muß noch an Pin 10 des 74HC4040 provisorisch ein Stück Kabel angelötet und das andere Ende in den Pin 7 der Oszillatorfassung auf der Hauptplatine gesteckt werden, damit der Rechner weiterhin seinen Master-Takt bekommt. Sonst (260/520/1040ST) ist ja der Hauptoszillator noch nicht angetastet worden.
- Als nächstes müssen Sie die Kabel wie folgt anschließen:
  - Die mit 'DS0' und 'DS1' bezeichnete Leitung kommt an Pin 19 resp. 20 des Soundchips ('YM2149' oder 'AY-3-8910', 40poliges IC).
  - Die Leitung 'HD' löten Sie bitte an Pin 2 des Shugart-Busses zur Floppy, falls Sie planen, ein HD-Laufwerk statt der alten Floppy zu nutzen. Stellen Sie provisorisch eine Brücke zu Pin 1 (Masse) her, denn bisher ist das Laufwerk kein HD-Laufwerk. Andernfalls führen Sie die Leitung nach außen zu Ihrem externen HD-Laufwerk (dort

Das HD-Interface

Die Schaltung des HD-Interfaces können Sie Bild 5 entnehmen. Sie beinhaltet keine Ungewöhnlichkeiten; abhängig vom Zustand der beiden Drive-Select-Leitungen und der HD-Detect-Leitung werden entweder 8 oder 16 MHz über einen Analogmultiplexer an den Floppycontroller gelegt. Im Ruhezustand liegen 8 MHz an, und nur bei Zugriff auf eine HD-Diskette wird hochgeschaltet auf 16 MHz. Der Zähler liefert noch die 2 MHz für den Soundchip und 500 kHz für die ACIAs.

Für Mischbetrieb eines normalen und eines HD-Laufwerks müssen Sie dem normalen Laufwerk die Fähigkeit nachrüsten, auf der HD-Detect-Leitung den Typ der eingelegten Diskette zu melden. Das läßt sich ganz einfach bewerkstelligen, indem man von der Drive-Select-Leitung des Laufwerks (am besten HINTER dem Jumper, mit dem man Betrieb als Laufwerk A oder B wählen kann!) aus eine Schottky-Diode (z.B. BAT 85) mit der Anode an die HD-Detect-Leitung (Leitung 2 des Shugart-Busses) legt (= Markierungsstrich an Drive-Select, siehe Bild 6). Damit wird die Leitung HD-Detect immer dann auf Low gezogen, wenn das Laufwerk selektiert wird, entsprechend geht der Floppycontroller völlig zu Recht von einer DD-Diskette aus. Der Vorteil dieser umständlich anmutenden Prozedur ist, daß die Floppies z.B. in einem Tower-Gehäuse ganz normal mit Flachbandkabel verdrahtet, auch ggf. gegeneinander ausgetauscht werden können, und für die HD-Floppies, die ja schließlich nichts für die Dummheit ihrer älteren Kollegen können, kein Zusatzaufwand nötig ist.

Für ausschließlichen Betrieb von normalen Floppies genügt es natürlich, 'HD' fest auf Masse zu legen.

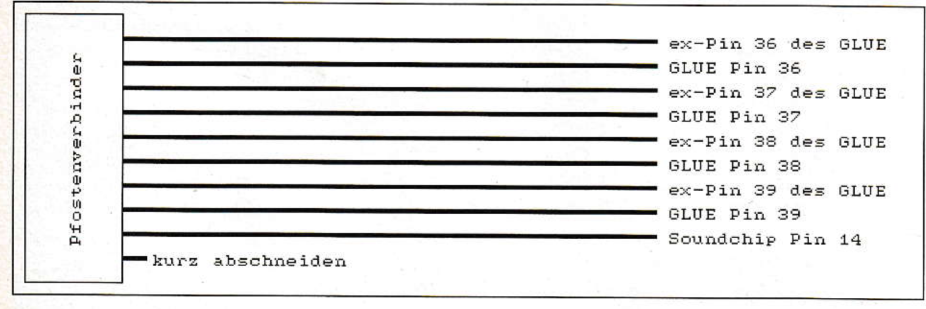


Bild 4: Anschlußschema EPROM-Sequencer

provisorisch auf Masse legen!), oder aber legen Sie sie (vorläufig?) schlicht irgendwo auf Masse.

- Die beiden Taktleitungen lassen Sie erstmal so baumeln, daß sie nichts unbefugt berühren können, denn wir machen wieder einen kleinen Zwischentest:

- In diesem Zwischenstadium (bezüglich der Sicherheitshinweise möchte sich der Autor nicht ständig wiederholen) zusammengebaut und eingeschaltet, sollte der Rechner ganz normal funktionieren. Erscheint jedoch nach dem Einschalten kein Bild, haben Sie entweder einen Kurzschluß gebaut (das Netzteil 'singt' dann), oder aber die 32MHz-Taktleitung zur Oszillatorfassung ist fehlerhaft. Ist Zugriff auf die Floppy unmöglich, ist vermutlich ein Fehler in der Frequenzteilungsschaltung, der mit Oszilloskop oder Frequenzzähler gesucht werden muß, oder aber die Leitung 'HD' liegt nicht auf Masse. (Siehe auch die erklärenden Hinweise zum HD-Adapter im nebenstehenden Textkasten 1, die hier zu weit führen würden).
- Im nächsten Schritt des Umbaus wird erstmals etwas nachhaltig zerstört: Sie müssen die Taktleitungen vom GLUE zum Soundchip (2MHz) und zu den ACIAs (500kHz) auftrennen. Dazu sucht man sich am geschicktesten vom Soundchip ausgehend (Pin 22) eine Durchkontaktierung und kratzt auf der anderen Seite HINTER der Durchkontaktierung die Leiterbahn (nur eine!!) durch. So kann man die Leitung '2MHz' von der Floppycontroller-Platine komfortabel in die Durchkontaktierung löten. Ebenso macht man es mit den beiden ACIAs '6850', sie sitzen unmittelbar benachbart, indem man vom jeweiligen Pin 3/4 (die sind beide zusammengeschaltet) bis zu einer Durchkontaktierung geht, wo die Taktleitungen beider ACIAs bereits zusammen sind, damit hinterher sowohl MIDI als auch die Tastatur normal weiterfunktionieren. Hier wird nun, Sie werden es erraten haben, die Leitung '500kHz' angelötet.
- Auch an dieser Stelle ist wiederum ein kurzer Test des Rechners angebracht.

Alles ist nun vorbereitet ...

... für die Versorgung mit mehr Taktfrequenz. Doch halt, die Anpassung der Bildschirmsynchronsignale fehlt noch. Daher müssen Sie zunächst die Leiterplatte nach Bild 2 bestücken. Dabei gelten ganz analoge Aufbauhinweise wie bei der Floppycontroller-Platine (für den Quarzoszillator werden vier Einzel-Pins in entsprechend große Löcher gelötet!). Für die HC(T)-Bausteine können die Sockel eingespart werden, man tut sich dann beim beidseitigen Verlöten der wenigen oben geführten Leitungen leichter (der HC4040 unter dem EPROM kann sowieso nicht gesockelt werden wegen der Bauhöhe). Es geht aber auch mit Fassungen mit gedrehten Kontakten. Das EPROM muß aus naheliegenden Gründen auf alle Fälle wechselbar sein. Die Stiftleiste kann aus Gründen der Bauhöhe gewinkelt sein, muß es aber nicht. Vergessen Sie die Durchkontaktierungen daneben nicht!

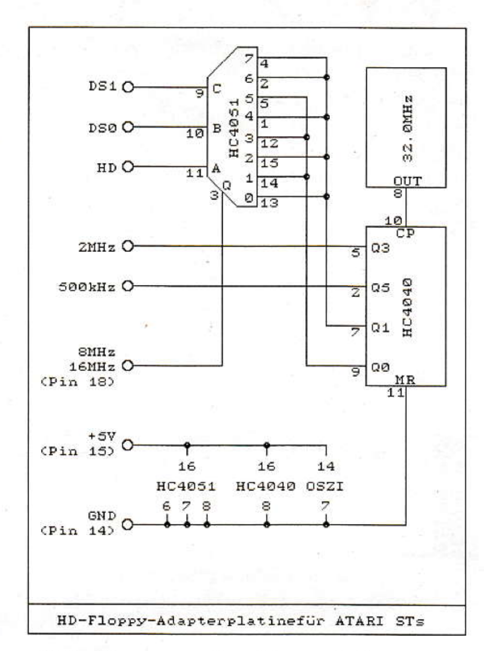


Bild 5: Floppycontroller-Adapterschaltung

Kurz zur Funktion der Schaltung in Bild 3 (die ganze Schaltung bezeichnet man als EPROM-Sequencer, sie ist für einfache Steuerungsaufgaben leicht zweckzuentfremden):

Der 'Master-Takt' von 32 - 48 MHz speist einen als Frequenzteiler durch 32 beschalteten 4024. Die aus diesem kommenden 1 - 1.5 MHz takten das Ausgangsregister, das zur Unterdrückung von Spikes, wie sie aus einem EPROM, dessen Adressen man bei aktiviertem Ausgang einfach hochzählt, zwangsläufig kommen, eingefügt ist. Eine Binärzählerkette, aufgebaut mit dem zweiten 4024 und einem 4040, steuert die Adreßleitungen A0 - A14 des EPROMs an. In den Datenwörtern ist die jeweilige Information abgelegt, ob z.B. VSYNC gerade aktiv oder inaktiv ist.

Die Datenleitung D6 kann durch vorzeitigen Reset der Zähler den Zyklus verkürzen, so daß in Verbindung mit geeigneten EPROM-Inhalten fast jedes beliebige Bildschirm-Timing programmierbar ist. Die Adreßleitung A15, von einem freien Port des Soundchips umgeschaltet, wählt zwischen zwei Auflösungen aus. Denkbar, aber bisher nicht getestet, wäre auch ein Auflegen der Monochrom-Detect-Leitung an A15, um weiterhin alternativ Farbe und Monochrom wie gewohnt zu haben.

Um die EPROM-Erzeugung kümmert sich das kleine GFA-Programm aus Listing 1. Es erzeugt, wahlweise für 40 oder 48 MHz Master-Takt, ein Standard-EPROM mit 640x400 Punkten in der unteren Speicherseite und dem Pixel-Takt entsprechend mehr Auflösung in der oberen Speicherseite.

Auf der Diskette zum Heft ist eine komfortablere Version des Programms enthalten, ebenso wie der weiter unten beschriebene Treiber für die höhere Auflösung als Quelltext ohne Abtippen. Bei der höheren Auflösung werden die Timing-Spezifikationen des SM 124 exakt eingehalten, was zu 800x400 Punkten bei 40 MHz und 960x400 Punkten bei 48 MHz führt. Da jedoch jeder dem Autor bekannte SM 124 mindestens 480 Zeilen darstellen kann, können Sie mit Hilfe der entsprechend kommentierten Stelle im Listing auch jeweils 480 Punktzeilen erzeugen. Der Modus mit der Originalauflösung in der unteren Speicherseite ist notwendig, um Meldungen von Autoordner-Programmen lesen zu können (nicht so wichtig) und insbesondere, weil der Bildspeicher direkt nach dem Einschalten nicht groß genug ist für mehr Punkte (das allerdings ist entscheidend!).

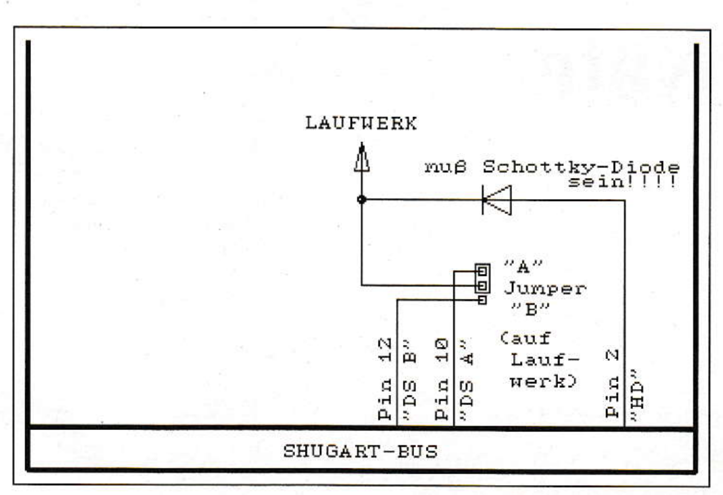


Bild 6: Modifikation älterer Laufwerke für Mischbetrieb mit HD-Laufwerken

Sie sollten die Platine fertig, also mit programmiertem EPROM, bestücken, jedoch noch NICHT einbauen!

... weiter basteln

Leider kann Ihnen weiteres Kratzen von Leiterbahnen auf der Hauptplatine nicht erspart werden. Doch zunächst pressen Sie einen 10poligen Pfostenverbinder an Flachbandkabel an und spleißen die Adern auf (Bild 4). Die Leitung 'Port' wird an Pin 14 des Soundchips gelötet. Mit den Leitungen 'DE', 'BLNK', 'HSYN' und VSYN' ist es komplizierter: Diese Signale kommen im Original vom GLUE (PLCC-IC links unterhalb des MEGA-Netzteils), von seinen Pins 36 - 39. Diese liegen gegenüber der Seite mit dem Punkt; die Pins sind vom Punkt aus (Pin 1) gegen den Uhrzeigersinn durchnumeriert (von oben gesehen). Von dort gehen im MEGA ST vier Leiterbahnen schräg weg zu vier Durchkontaktierungen. (An dieser Stelle sei den Besitzern der unzähligen Varianten von 1040er-Platinen, aber auch den 260/520ST-Eignern, viel Erfolg beim Suchen entsprechender Durchkontaktierungen gewünscht, nachdem sie überhaupt erst mal den GLUE gefunden haben!). Diese Leiterbahnen werden durchtrennt. Nun wird folgendermaßen neu verdrahtet:

- An die Durchkontaktierung, die mit Pin 36 verbunden war, kommt die Leitung 'BLNK'. Die benachbarte Litze, die, die auf der Stiftleiste an den gegenüberliegenden unbenutzten Stift führt, wird am GLUE-Sockel an Pin 36 angelötet.
- Analog schließen Sie Pin 37 ('HSYN'), Pin 38 ('VSYN') und Pin 39 ('DE') an.

Sinn dieser Art der Verdrahtung ist, daß man sich aus einem Stück Stiftleiste 2x4 einen 'Bypass' bauen kann, der jeweils gegenüberliegende Pins verbindet. Ein mit durchkratzten Leiterbahnen, angelötetem Kabel und eingestecktem 'Bypass' probehalber in Betrieb genommener Rechner sollte weiterhin normal funktionieren!

Spannend ...

... wird es beim nächsten Schritt. Die Bildschirm-Timing-Adapterplatine wird eingebaut. Das ist im MEGA ST ganz einfach: Die Sockel-Pins des Oszillators kommen in den Oszillatorsockel auf dem Motherboard, dabei zeigt die Stiftleiste zum GLUE.

In den alten 260STs ist im Shifter-Blechkasten auf der Platine ein unbestückter Platz für einen 14poligen Sockel für die Modulatorschaltung. Hier kann man einen Sockel nachbestücken und mittels einiger Zwischensockel wegen des Abschirmkastens den EPROM-Sequencer aufstecken. Danach muß man noch Pin 8 dieser Fassung (40 - 48 MHz Taktausgang) mit Pin 2 des Shifters verbinden, nachdem man die Originalzuleitung zu Pin 2 durchgekratzt hat.

Für andere ST-Modelle kann mangels konkreter Erfahrung keine detaillierte Einbauanleitung gegeben werden; entscheidend ist, die Leiterplatte mit Pin 2 des Shifters (Pin 8 des Oszillators), mit Masse (Pin 7 des Oszillators) und mit +5V (entsprechend Pin 14) zu verbinden. Alle Verbindungen, insbesondere Masse, kurz und dick, besser noch mehradrig, ausführen!!

Fürs erste Einschalten läßt man den 'Bypass' im Kabel und zieht UNBEDINGT DEN MONITOR ab. Der Rechner sollte nun ganz normal booten, bloß daß man nichts davon sieht. Er läuft jetzt bereits mit dem höheren Takt. Nun ist Gelegenheit, mit einem Oszilloskop bzw. Frequenzzähler die Bildschirmsynchronsignale des Sequencers zu kontrollieren: 'VSYN' muß mit 71.3 Hz stabil anliegen, 'HSYN' mit 35.7 kHz und die Signale 'BLNK' und 'DE' ebenfalls, wobei letztere gelegentliche Aussetzer haben, da sie nicht immer aktiv sind während eines Vollbildes (siehe auch erklärender Textkasten 2). Nur wenn Sie sich sicher sind, daß die Signale ok sind, dürfen Sie den Rechner ausschalten, den Bypass entfernen und das Flachbandkabel an den Sequencer anschließen. Beim nächsten Einschalten sollte der Rechner normal funktionieren, bloß nach wie vor ohne Bild. Nun sollten Sie den Monitor anstecken. Der Rechner wird neu booten, denn kein Monitor ist für den ATARI gleichbedeutend mit 'Farbmonitor angeschlossen', und sollte dabei ein weißes Rechteck, wie gewohnt, nur etwas schmaler, zeigen.

Ausgefranste Ränder deuten auf fehlerhafte Synchronsignale oder nicht quarzstabilen Takt hin. Hellgetastete Zeilen im Strahlrücklauf bedeuten Fehler in den 'DE' und 'BLNK'-Signalen. Bildstörungen (insbesondere unmotiviert auftauchende Pünktchen ...) weisen leider auf zu langsames RAM hin. Daher werden Sie auf 10 MHz zurückgehen oder schnellere RAMs einlöten müssen.

Ist bis hierher alles glatt gegangen, sollten Sie 'NEWVIDEO.S' abtippen, assemblieren und in den Autoordner Ihrer Boot-Partition (-Diskette) kopieren. Die Funktion dieses Programms entnehmen Sie bitte den üppigen Kommentaren. Es funktioniert am besten als das physikalisch letzte Autoordner-Programm, was sich zunächst durch das einfache Einkopieren auch automatisch einstellt. Wenn alles richtig abgetippt, insbesondere die Konstanten im Programmkopf richtig gesetzt sind, das EPROM den passenden Inhalt hat und alles richtig verdrahtet ist (letzteres wurde immer wieder zwischendurch überprüft), sollte beim nächsten Einschalten nach dem Abarbeiten des Autoordners der vergrößerte Bildschirm erscheinen.

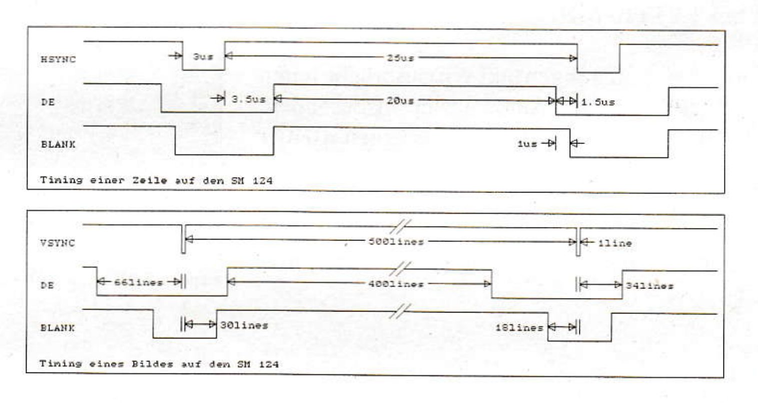


Bild 7: Soll-Timing des SM 124

Wie bei allen anderen dem Autor bekannten Bildschirmvergrößerungen muß man auch hier mit ein paar Nachteilen leben:

- Manche Programmierer halten nichts von mehr Bildschirmfläche (eine aussterbende Spezies).
- Der VBLANK-Interrupt kommt allenfalls zufällig mal im tatsächlichen Bildrücklauf, weil er nach wie vor vom GLUE erzeugt wird. Daher flimmern Animationen auf einmal so schön. Dieses Problem hat diese Bildschirmerweiterung allerdings gemeinsam mit vielen anderen Grafikkarten bzw. Erweiterungen

Zum guten Schluß ...

... noch ein paar Hinweise: Wenn Sie aus der Monatsdiskette das komfortable EPROM-Konfigurationsprogramm benutzen, denken Sie immer daran, Ihren Monitor nicht zu überlasten. Der kritischste Parameter ist wegen der Kopplung der Hochspannungserzeugung an die Zeilen-

frequenz die Horizontalfrequenz. Sie sollte NIE unterschritten (der Monitor synchronisiert dann nicht) und NIE überschritten (die Hochspannung steigt unkontrolliert an) werden. Wesentlich unkritischer ist die Vertikalfrequenz; senken Sie sie ruhig etwas ab gegenüber den 71.3 Hz, das ergibt ein paar Bildzeilen. Irgendwann jedoch wird das Bild durchlaufen wie bei einem defekten Fernseher, dann haben Sie es übertrieben! Sind Sie jedoch Besitzer eines Multiscan-Monitors, können Sie weggehen von den starren Vorgaben 35.7 kHz/71.3 Hz hin zur Leistungsgrenze des Monitors (meist 38 kHz) und zur Flimmergrenze der Augen (individuell unterschiedlich). Wichtig ist dabei noch, daß ein normaler Multiscan auch nur einen 10 MHz-ATARI ohne neues Bildschirm-Timing NICHT verträgt, da 10 MHz/8 MHz x 35.7 kHz mehr als die üblichen 38 kHz ergibt!

Wer auf die Bildschirmvergrößerung keinen Wert legt, kann auch nur die Originalauflösung in ein 256er-EPROM brennen und die Leitung 'port' unbeschaltet lassen. Genauso können auch Besitzer von MEGASCREEN (!) oder PIXELWONDER (?) verfahren, um ihren Rechner weiterhin mit dieser Grafikerweiterung nutzen (sprich: einschalten, denn dann sind diese Schaltungen noch nicht aktiv) zu können. Beim Erstellen von Monitoranpassungen müssen Sie aber UNBEDINGT beachten, daß die Behauptungen der Konfigurationsprogramme zu Horizontal- und Vertikalfrequenz um 1.25 (10 MHz) bzw. 1.50 (12 MHz) ZU NIEDRIG liegen!!

Hinweise, Erfahrungen, Anregungen und Wutausbrüche leiten Sie bitte über die Redaktion an den Autor weiter, insbesondere wären ergänzende Hinweise über einen erfolgreichen 1040er Umbau sicherlich für viele Leser interessant.

Abschließend noch der Hinweis, daß die in der vorangegangenen Artikelserie vorgestellten Hardware-Lösungen kommerziell ohne Zustimmung von Redaktion und Autor nicht verwertet werden dürfen.

Michael Steinle

STÜCKLISTE

1	MC 68000-10 bzw. -12
1	Quarz-Oszillator 40.000 MHz oder 48.000 MHz
1	Quarz-Oszillator 32.000 MHz (nur für 260/520/1040ST)
2	74HC(T)4040 (Sequencerplatine, Floppycontrolleradapter)
2	74HC(T)4024 (Sequencerplatine)
1	74HC(T)4051 (Floppycontrolleradapter)
1	74HC(T)574 (Sequencerplatine)
2	74ALS244 oder 74F244 (Motherboard)
2	74ALS373 oder 74F373 (Motherboard)
1	27C512 oder ggf. 27C256 (siehe Text)
1	4k7 (Sequencerplatine)
1	1k (Sequencerplatine)
1	47p (Sequencerplatine)
2	100n (Sequencerplatine)
2	560 (Motherboard)
2	10kx8 Netzwerke (nur MEGA ST, Motherboard)
2	4k7x9 Netzwerke (nur 260/520ST, Motherboard)
	alternativ 2x3k3x8 resp. 2x3k3x9 für fleißige Leute
2	28polige Fassung (Sequencerplatine, Motherboard)
4	20polige Fassung (Motherboard)
1	64polige Fassung (Motherboard)
36	Einzelpins, ggf. aus 40poliger Fassung gewinnen!
1-3	14polige Fassungen (Motherboard, Zwischensockel)
1	Pfostenleiste 2x5, ggf. abgewinkelt
1	Pfostenverbinder 10polig
	etwas Flachbandkabel und normales Kabel
	viel Zeit und Geduld

```
'Programm erzeugt EPROM-Dateien
'(c)1992 by MAXON-Computer
'Autor: Michael Steinle
'
DEFWRD "a-z"
DIM eprom|(32767)
eprom$="NEWVIDEO.BIN"
DEFMOUSE 2
' ein EPROM für SM124 enthält generell 501 Pixel-
' zeilen der Original-Zeilenlänge. Die Brutto-
' Zeilenlänge beträgt bei 8 MHz 896 Pixel, bei
' 10 MHz 1120 Pixel und bei 12 MHz 1344 Pixel
' make_eprom(96,640,64,96,66,400,34,1)
'                                ! 640x400 bei  8 MHz
make_eprom(256,640,96,128,66,400,33,2)
'                                ! 640x400 bei 10 MHz
' make_eprom(384,640,160,160,66,400,33,2)
'                                ! 640x400 bei 12 MHz
write_eprom                      ! untere EPROM-Hälfte
'
' make_eprom(64,704,32,96,66,400,34,1)
'                                ! 704x400 bei  8 MHz
' make_eprom(64,704,32,96,18,480,2,1)
'                                ! 704x480 bei  8 MHz
make_eprom(128,800,64,128,66,400,33,2)
'                                ! 800x400 bei 10 MHz
' make_eprom(128,800,96,64,128,18,480,1,2)
'                                ! 800x480 bei 10 MHz
' make_eprom(160,960,64,160,66,400,33,2)
'                                ! 960x400 bei 12 MHz
' make_eprom(160,960,64,160,18,480,1,2)
'                                ! 960x480 bei 12 MHz
append_eprom                     ! obere EPROM-Hälfte
DEFMOUSE 0
EDIT
'
PROCEDURE write_eprom            ! EPROM-Inhalt unter
  OPEN "O",#1,eprom$             ! 'eprom$' ins aktu-
  BPUT #1,V:eprom|(0),32768 ! elle Verzeichnis
  CLOSE #1                       ! schreiben
RETURN
'
PROCEDURE append_eprom           ! EPROM-Inhalt an
  OPEN "U",#1,eprom$             ! bestehende Datei
  SEEK #1,32768                  ! 'eprom$' anhängen
  BPUT #1,V:eprom|(0),32768
  CLOSE #1
RETURN
'
PROCEDURE make_eprom(abh,hs,bish,h,abv,vs,bisv,v)
  ' wegen EPROM-Auflösung von 32pixel/Schritt
  ' folgende Umrechnungen:
  abh=abh DIV 32     ! unsichtbarer Teil jeder
  '                  ! Zeile nach HSYNC in Pixel
  hs=hs DIV 32       ! sichtbarer Teil jeder Zeile
  '                  ! in Pixel
  bish=bish DIV 32 ! unsichtbarer Teil jeder
  '                  ! Zeile bis HSYNC in Pixel
  h=h DIV 32         ! Dauer von HSYNC in Pixel
  ' abv              ! unsichtbarer Teil des Bildes
  '                  ! nach VSYNC in Zeilen
  ' vs               ! sichtbarer Teil des Bildes
  '                  ! in Zeilen
  ' bisv             ! unsichtbarer Teil des Bildes
  '                  ! bis VSYNC in Zeilen
  ' v                ! Dauer von VSYNC in Zeilen
  '
  LOCAL bildhoehe,zeilenbreite,i,j,k,filler|
  LOCAL de,blnk,vsyn,hsyn
  '
  hsyn=0             ! Bit-Nummern im EPROM
  vsyn=1
  de=2
  blnk=3             ! low-aktiv!
```

→

ATARI–Kompetenz im Abo !

Der Inhalt

Wir recherchieren für Sie:

- Aktuelle Meldungen aus der ATARI–Szene
- Hintergrundwissen, damit Sie rundum informiert sind und mitreden können
- Ausführliche Hardware und Software–Tests der neuesten Produkte
- Programmierer Tips & Tricks von Lesern für Leser





Der Preisvorteil

Sie erhalten 11 Ausgaben zum Preis von 10. Sie zahlen also nur 80,– DM statt 88,– DM im Einzelheftverkauf.



Der Zustellkomfort

Sie erhalten die ST–Computer per Post frei Haus geliefert und ersparen sich so den Weg zum Kiosk.

Der Bonus

1.) Sie erhalten eine Sonderserien–Diskette aus dem Atari–Journal
2.) Sie erhalten den aktuellen Public–Domain Katalog
Beide Zugaben sind im Abo–Preis inbegriffen.



Der Steuer–Spartip

Die ST–COMPUTER ist eine Fachzeitschrift. Dadurch können Sie Ihr Abonnement der ST–Computer steuerlich geltend machen.



Die Disketten zum Heft

Alle Listings gibt's auch auf Diskette zum Preis von 12,– DM. Auf der Diskette sind jeweils 2 Monate zusammengefaßt.

Bitte benutzen Sie die Bestellkarte in der Heftmitte.

```
  filler|=BSET(filler|,hsyn)
  '                  ! HSYNC normal inaktiv high
  filler|=BSET(filler|,vsyn)
  '                  ! VSYNC ebenfalls
  ARRAYFILL eprom|(),filler|
  ' EPROM-Inhalt vorbesetzen
  '
  zeilenbreite=abh+hs+bish+h
  bildhoehe=abv+vs+bisv+v
  '
  FOR i=0 TO bildhoehe-1
    k=i*zeilenbreite
    FOR j=0 TO zeilenbreite-1
      IF j>=abh AND j<abh+hs
        ' horizontal sichtbar
        IF i>=abv AND i<abv+vs
          ' vertikal sichtbar
          eprom|(k+j)=BSET(eprom|(k+j),de)
          eprom|(k+j)=BSET(eprom|(k+j),blnk)
          ' DE setzen und BLNK inaktiv machen
        ENDIF
      ELSE IF j>=zeilenbreite-h
        ' während HSYNC-Zeit HSYNC setzen
        eprom|(k+j)=BCLR(eprom|(k+j),hsyn)
      ENDIF
    NEXT j
    IF i>=bildhoehe-v
      ' während VSYNC-Zeit
      FOR j=0 TO zeilenbreite-1
        eprom|(k+j)=BCLR(eprom|(k+j),vsyn)
        ' die ganze Zeile lang VSYNC setzen
      NEXT j
    ENDIF
  NEXT i
  k=k+j
  ' erstes EPROM-Byte 'hinter' dem Bild
  ' Rest des EPROMs mit $FF füllen:
  WHILE k<32767
    eprom|(k)=&HFF
    INC k
  WEND
RETURN
```

```
* Treibersoftware für Nutzung des EPROM-
* Sequencers zur Bildschirmvergrößerung. Die dort
* programmierte Auflösung muß bei den Labels am
* Anfang des Programms eingetragen werden:
* (c)1992 by MAXON-Computer
* Autor: Michael Steinle

anfang:     bra.w     start
hor_res:    dc.w      768
ver_res:    dc.w      544

screenlen:  ds.l      1
physbase:   ds.l      1
back:       ds.l      1
intout:     ds.l      1

            dc.b      "XBRA"
            dc.b      "NEWV"
oldtrap2:   dc.l      0
mytrap2:    cmp.w     #$73,d0         ;VDI?
            bne.s     normtrap2       ;nein, also
*                         AES; -> Originalroutine
            move.l    d1,a0           ;Zeiger auf
*                           Parameterblock in a0
            move.l    12(a0),intout   ;Zeiger auf
*                             Intout-Feld retten
            move.l    (a0),a0         ;Zeiger auf
*                            Control-Feld holen
            cmp.w     #1,(a0)         ;Funktions-
*                          nummer 1 (OPN_WK)?
            bne.s     normtrap2       ;nein, also
*                            -> Originalroutine
            move.l    2(sp),back      ;sonst Rück
*                          sprungadresse retten
            move.l    #patch,2(sp)    ;und auf
*                    dem Stack herummanipulieren
normtrap2:  move.l    oldtrap2,-(sp)  ;Original-
*                             routine aufrufen
            rts

* Beim Rücksprung von V_OPNWK wird der folgende
* Programmteil durchlaufen:

patch:      dc.w      $A000           ;LINE-A-
*                      Initialisierung aufrufen
            move.l    intout(pc),a1   ;Zeiger auf
*                        Intout-Tabelle zurück

            move.w    ver_res(pc),d0  ;vertikale
*                             Auflösung holen
            move.w    d0,-4(a0)       ;vertikale
*                   Auflösung in LINE-A einpatchen
            subq.w    #1,d0
            move.w    d0,-$2B2(a0)    ;auch in
*                 Workstation-Tabelle eintragen
            move.w    d0,2(a1)        ;auch in
*                               INTOUT patchen

            move.w    -$2E(a0),d1     ;Zeichen-
*                                  höhe holen
            divu      d1,d0           ;vertikale
*                   Auflösung durch Zeichenhöhe
            move.w    d0,-$2A(a0)     ;als Zei-
*                   lenzahl-1 (VT 52) eintragen

            move.w    hor_res(pc),d0  ;horizon-
*                       tale Auflösung holen
            move.w    d0,-$C(a0)      ;in LINE-A
*                                  einpatchen
            subq.w    #1,d0
            move.w    d0,-$2B4(a0)    ;auch in
*                       die Workstation-Tabelle
            move.w    d0,(a1)         ;und in
*                                      INTOUT
            addq.w    #1,d0           ;wird
*               wieder echte Pixelzahl gebraucht:

            lsr.w     #3,d0           ;horizon-
*                   tale Auflösung durch 8
            subq.w    #1,d0
            move.w    d0,-$2C(a0)     ;als Spal-
*                   tenzahl-1 (VT 52) eintragen
            addq.w    #1,d0           ;echte
*                                 Spaltenzahl
            move.w    d0,-2(a0)       ;als Bild-
*                            breite in Bytes
            move.w    d0,2(a0)        ;gleich
*                     nochmal (!!) einpatchen
            mulu      d1,d0           ;x Zeichen-
*                      höhe = Bytes/Charzeile
            move.w    d0,-$28(a0)     ;in LINE-A
*                                   eintragen

* vergrößerten Bildschirm löschen von den Mel-
* dungen der vorangegangenen Autoordnerprogramme

            move.l    physbase(pc),a0;Bildspei-
*                           cheranfang holen
            move.l    screenlen,d0    ;soviele
*                          Langworte löschen
            subq.l    #1,d0           ;wegen DBRA
erase:      clr.l     (a0)+
            dbra      d0,erase

* ab hier wird auf die höhere Auflösung umge-
* schaltet mit Hilfe von Bit Nr. 7 des Soundchips
* (bisher unbenutzt, beim Booten auf 0)

            move.w    sr,-(sp)        ;Status-
*                          register sichern
            or.w      #$700,sr        ;Interrupts
*                                 abschalten
            lea       $FF8800,a0      ;Zeiger auf
*                                  Soundchip
            move.b    #$E,(a0)        ;Port A
            move.b    (a0),d0         ;auslesen
            or.b      #$80,d0         ;setze Bit7

* die folgende Routine ist aus TOS 1.04 abge-
* schrieben. Ohne Synchronisierung auf VSYNC
* geht das Umschalten des Bildschirms schief!

            movem.l   d0-d4/a0-a1,-(sp)
            lea       $FFFFFA21,a0    ;TIMER B
            lea       $FFFFFA1B,a1
            move.b    #$10,(a1)
            moveq     #1,d4
            move.b    #$0,(a1)
```

→

```
                move.b      #$F0,(a0)
                move.b      #$8,(a1)
wait1:          move.b      (a0),d0
                cmp.b       d4,d0
                bne.s       wait1
wait2:          move.b      (a0),d4
                move.w      #600,d3          ;falls Stö-
*         rungen auftreten, mit diesem Wert spielen!
wait3:          cmp.b       (a0),d4
                bne.s       wait2
                dbra        d3,wait3
                move.b      #$10,(a1)
                movem.l     (sp)+,d0-d4/a0-a1

                move.b      d0,2(a0)         ;Port A mit
*                 gesetztem Bit 7 zurückschreiben
                move.w      (sp)+,sr         ;Statusre-
*                             gister zurückholen
                move.l      back(pc),-(sp)   ;zurück-
*                                    springen
                rts

* Programminitialisierungsteil:

start:          move.w      #-1,-(sp)
                move.w      #11,-(sp)        ;Status der
*                          Umschalttasten holen
                trap        #13
                addq.l      #4,sp
                andi.l      #4,d0            ;Control
*                                    gedrückt?
                bne         abflug           ;ja, daher
*                              nix instalieren

                move.l      4(sp),a6         ;Basepage-
*                                     adresse
                move.l      #$100,d5         ;Länge der
*                                    Basepage
                add.l       $C(a6),d5        ;+ Länge
*                                   des Codes
                add.l       $14(a6),d5       ;+ Länge
*                         des Konstantenbereichs
                add.l       $1C(a6),d5       ;+ Länge
*                                     des BSS
                lea         stack(pc),a7     ;eigener
*                                Stack bitte

                moveq       #0,d7
                move.w      hor_res(pc),d7   ;horizon-
*                          tale Auflösung holen
                lsr.w       #5,d7            ;und durch
*                           32 (Longs!) teilen
                muls        ver_res(pc),d7   ;mal verti-
*                           kale Auflösung nehmen
                move.l      d7,screenlen     ;für Lösch-
*                           routine abspeichern

                move.l      d5,-(sp)         ;NEWSIZE
                move.l      a6,-(sp)         ;BLOCKADR
                move.w      #0,-(sp)         ;Nullparam.
                move.w      #74,-(sp)        ;MSHRINK
                trap        #1               ;nicht be-
*                     nötigten Speicher freigeben
                lea         12(sp),sp

                asl.l       #2,d7            ;d7 * 4 =
*                     Bildschirmspeicher in Bytes
                sub.l       #32000,d7        ;das ist
*                          der zusätzliche Bedarf
                asr.l       #8,d7            ;zusätz-
*              lichen Bedarf in Vielfache von 256
                asl.l       #8,d7            ;umrechnen
*            wegen 256-Byte-Grenzen des Videoteils

                move.l      #-1,-(sp)        ;wieviel
*                             Speicher ist da?
                move.w      #72,-(sp)        ;MALLOC()
                trap        #1
                addq.l      #6,sp
                sub.l       d7,d0            ;Bildspei-
*                             cherbedarf abziehen
                bmi         error1           ;zuwenig
*                      Speicher vorhanden (????)

                move.l      d0,-(sp)         ;alles
*                            übrige reservieren
                move.w      #72,-(sp)        ;MALLOC()
                trap        #1
                addq.l      #6,sp
                move.l      d0,d5            ;Anfangs-
*                             adresse aufheben

                move.l      d7,-(sp)         ;Bildspei-
*                   cher-Zusatzbedarf reservieren
                move.w      #72,-(sp)        ;MALLOC()
                trap        #1
                addq.l      #6,sp
                move.l      d0,d4            ;für Plau-
*                     sibilitätsprüfung aufheben!

                move.l      d5,-(sp)         ;Adresse
*                                 erster Block
                move.w      #73,-(sp)        ;diesen
*                         Block wieder freigeben
                trap        #1
                addq.l      #6,sp

                move.w      #2,-(sp)         ;noch aktu-
*                      elle PHYSBASE ermitteln
                trap        #14
                addq.l      #2,sp

                sub.l       d7,d0            ;Zusatz-
*                        speicherbedarf abziehen
                cmp.l       d0,d4            ;sollte mit
*                       reservierter Blockgrenze
                bne         error            ;überein-
*                                    stimmen!

                move.l      d0,physbase      ;für später
*                        zum löschen aufheben!
                move.w      #-1,-(sp)        ;Bild-
*                   schirmspeicher vorverlegen
                move.l      d0,-(sp)
                move.l      d0,-(sp)
                move.w      #5,-(sp)         ;SETSCREEN
                trap        #14
                lea         12(sp),sp

                pea         message1(pc)
                move.w      #9,-(sp)
                trap        #1
                addq.l      #6,sp

                pea         install(pc)      ;Vektor-
*                   biegen im Supervisormodus
                move.w      #38,-(sp)
                trap        #14
                addq.l      #6,sp

                clr.w       -(sp)            ;OK zurück
                pea         256+start-anfang
                move.w      #$31,-(sp)
                trap        #1               ;PTERMRES

abflug:         pea         message2(pc)
                move.w      #9,-(sp)
                trap        #1
                addq.l      #6,sp
                clr.w       -(sp)            ;sang- und
*                       klangloses Programmende
                trap        #1

error:          move.l      d4,-(sp)         ;Adresse
*                                 zweiter Block
                move.w      #73,-(sp)        ;diesen
*                         Block wieder freigeben
                trap        #1
                addq.l      #6,sp
error1:         pea         message3(pc)
                move.w      #9,-(sp)
                trap        #1
                addq.l      #6,sp
                clr.w       -(sp)            ;sang- und
*                       klangloses Programmende
                trap        #1

install:        move.l      $88,oldtrap2
                move.l      #mytrap2,$88
                rts

                SECTION DATA

message1:       dc.b        27,"E",13,10," Bildschirm"
                dc.b        "vergrößerung wird akti"
                dc.b        "viert!",13,10,10,0
message2:       dc.b        13,10," Keine Bildschirm"
                dc.b        "vergrößerung erwünscht?!"
                dc.b        13,10,10,0
message3:       dc.b        13,10," Bildschirmver"
                dc.b        "größerung konnte nicht"
                dc.b        "in-",13,10," stalliert"
                dc.b        " werden wegen Speicher"
                dc.b        "problemen",13,10,10,0

                SECTION BSS

ok:             ds.w        1
                ds.l        512
stack:          ds.l        1
```

Chiptuning

Seit November 1991 ist TOS 2.06 erhältlich, das sowohl auf einem ST als auch auf einem STE lauffähig ist. Angeblich soll man dieses TOS, das mehr Komfort bietet, direkt in einen 1040 STE einsetzen können, doch leider ist es anscheinend bei einigen (oder vielen?) 1040 STE mit einfachem Einstecken der neuen TOS 2.06-EPROMs nicht getan.

TOS 2.06 im 1040 STE

Diese Erfahrung mußte ich jedenfalls machen, als ich den 1040 STE eines Bekannten mit den TOS 2.06-Chips ausrüsten wollte, die er sich gekauft hatte. Die alten TOS 1.06-Chips hatten zwar weniger Pins, da aber auf der Platine passende Sockel vorhanden waren und die Pin-Belegung der Stromversorgung stimmte, baute ich die Chips einfach ein und startete den Computer. Der Bildschirm und die Floppy-LED blieben aber wie erwartet dunkel, und es tat sich nichts. Sobald ich wieder TOS 2.06 einsetzte, lief alles wie vorher. Da die TOS 2.06-EPROMs in meiner TOS Extension Card (für normale STs) funktionierten, also nicht kaputt waren bzw. von mir kaputt gemacht worden waren, forschte ich nach den Grund, warum sie ihren Dienst im STE verweigerten, und fand ihn auch bald.

Problem(lösung)

Das Problem liegt darin, daß TOS 2.06 auf EPROM-Chips (Typ 27010, 32 Pins) ausgeliefert wird, während für TOS 1.06 ROM-Chips verwendet werden. Diese haben vier Pins weniger, die Adreßleitung A16 wird an einem anderen Pin erwartet, und das Signal /OE ist überhaupt nicht vorhanden. Der Inhalt der EPROMs kann also gar nicht ausgelesen werden, deshalb nimmt der Computer seine Arbeit nicht auf (kein Betriebssystem). Da aber 32polige Sockel vorhanden waren, versuchte ich, eine Lösung zu finden, ohne viel Aufwand doch noch TOS 2.06 zum Laufen zu bringen.

Die Pin-Belegung der Sockel stimmt, wie bereits oben gesagt, im Orginalzustand bis auf A16 und /OE mit der der EPROMs überein. A16 liegt nicht an Pin 2, sondern an Pin 24 an, wo /OE liegen müßte, das komplett weggelassen wurde, weil es sowieso nur mit Masse verbunden sein müßte. Dadurch, daß A16 an den Platz von /OE gelegt wurde, benötigt der ROM-Chip nur 28 statt 32 Pins, um alle nötigen Signale zu erhalten. Um nun EPROMs einbauen zu können, müßte also das Adreßsignal von Pin 24 auf Pin 2 verlegt und Pin 24 mit Masse verbunden werden. Man könnte hingehen und für diese Änderung ein paar Leiterbahnen durchtrennen und anders verbinden, aber es geht noch eleganter und einfacher.

Ataris Layout-Abteilung hat nämlich ein paar Lötbrücken spendiert, die es erlauben, diese Signale im Prinzip durch einfaches Umlöten zweier Nullohm-Widerstände (werden anstelle von Draht benutzt, weil die Bestückungsmaschinen damit besser zurechtkommen) an die entsprechenden Pins anzulegen, so daß EPROMs eingesetzt werden können und TOS 2.06 lauffähig wird (so was ähnliches gab's schon mal in [1]).

Vor dem Umbau

Bevor Sie loslaufen, um die TOS-Chips zu kaufen, lesen Sie bitte diesen Artikel zuende und denken daran, daß Sie durch den Umbau alle Gewährleistungs- und Garantieansprüche verlieren und der Umbau auf eigene Gefahr geschieht. Weder ich noch die Zeitschrift ST Computer haften für den Fall, daß Ihr Computer nach dem Umbau funktionsuntüchtig ist, weil dabei etwas kaputt gemacht wurde. Falls Sie den Umbau dann noch durchführen wollen, schauen Sie bitte zuerst in Ihrem Computer nach, ob auch alles so ist, wie ich es hier beschrieben habe. Da Atari zumindest bei den ST-Serien mehrfach die Platinen-Layouts geändert hat, kann ich das für die STE-Serie nicht ausschließen. Dabei könnten zum Beispiel die Lötbrücken weggefallen/umbenannt oder die Sockel auf 28 Pins reduziert worden sein, was den Umbau zwar erschweren, aber für etwas geübte Bastler nicht unmöglich machen würde. Sollte alles wie beschrieben vorhanden sein, können Sie den Umbau ohne Änderungen durchführen, ansonsten lesen Sie noch den Abschnitt „Umbauhindernisse".

Außerdem gibt es eine kleine Unsicherheit, was die Kennzeichnung der neuen Chips angeht. Als einziges Unterscheidungsmerkmal für LOW und HIGH (wichtig beim Einbau, siehe unten) ist mir die Bezeichnung ‚E0' und ‚EE' aufgefallen, wobei E0 für LOW und EE für HIGH steht. Daher kann es sein, daß der Computer nach dem Einbau nur deswegen nicht funktioniert, weil die beiden Chips in den falschen Sockeln stecken und nur getauscht werden müßten. Im Zweifelsfall bzw. wenn diese Bezeichnung fehlt, kann man allerdings probieren, man macht nichts kaputt, solange man die Chips richtig herum in den Sockel steckt (Kerbe des Chips normalerweise deckungsgleich mit Kerbe des Sockels), sonst kann man sie meistens wegwerfen.

Der Umbau

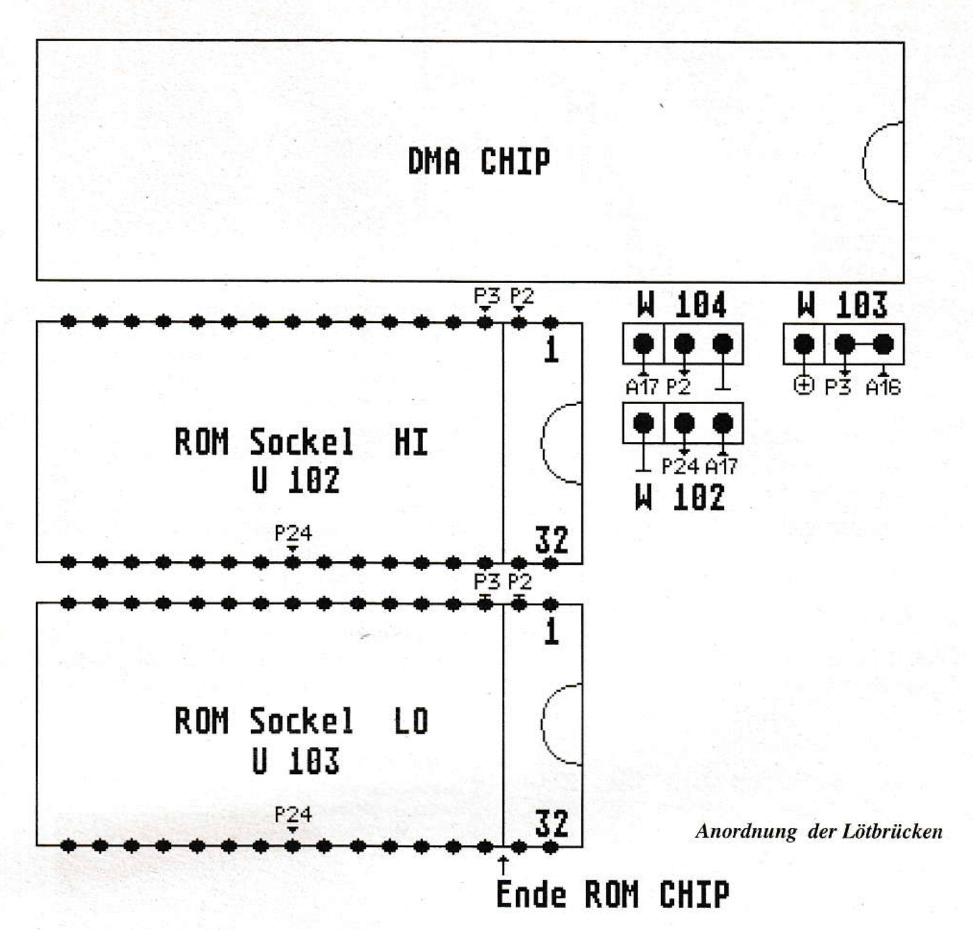
Sorgen Sie für eine saubere Arbeitsfläche, auf der man bequem arbeiten kann, und für einen brauchbaren Lötkolben (oder Lötstation). Ferner sollte ein Gefäß für die Schrauben und andere Kleinteile bereitstehen, damit nichts herrumrollt und verlorengehen kann. Noch zwei Hinweise: Passen Sie während des ganzen Umbaus auf, daß die Platine nicht stark gebogen oder belastet wird, da sonst Haarrisse (feine Unterbrechungen der Leiterbahnen) entstehen, die den Computer funktionsunfähig machen können und schwer zu finden sind (wenn überhaupt).

Als zweites sollten Sie mit dem Netzteil vorsichtig umgehen. Auch wenn es von der Stromversorgung getrennt ist, kann es noch gefährliche Spannungen führen. Fassen Sie also NIEMALS Bauteile usw. an, sondern halten Sie es nur am Blech fest. Falls Sie keine Diskette haben, von der man testweise booten kann, machen Sie sich noch eine zurecht, bevor Sie den Computer von der Stromversorgung trennen und auseinandernehmen. Er muß soweit zerlegt werden, daß die Platine alleine liegt (Tip: Um das untere Abschirmblech der Platine abnehmen zu können, müssen die kleinen Mutterschrauben an den Sub-D-Buchsen abgeschraubt werden).

Wenn Sie die Platine alleine vor sich liegen haben, werden Sie (hoffentlich) feststellen, daß die ROM-Chips in 32poligen Sockeln stecken (auf der Platinenseite, wo die Floppy montiert war, über dem Tastaturanschluß). Rechts neben den ROMs befinden sich die Lötbrücken W102, W103 und W104 (siehe Bild 1). Falls es jemand wundert, warum an den Brücken die Adreßleitungen immer eine Nummer größer sind: Die Adreßleitungen der Sockel sind direkt mit denen des Prozessors verbunden. Allerdings hat dieser keine Adreßleitung A0, so daß seine A1 mit der A0 des Sockels, seine A2 mit der A1 des Sockels usw. verbunden ist. Zurück zu den Lötbrücken: Nur W102 und W104 sind für uns interressant, W103 darf nicht verändert werden. Mit ihr könnte man anstatt des Adreßsignals A15 (A16 CPU) die Betriebsspannung 5V an Pin 3 der Sockel legen. A15 wird aber zum Adressieren des Chipinhalts benötigt, daher ist mir der Sinn dieser Brücke schleierhaft.

Löten Sie die beiden Widerstände, die jeweils das rechte und das mittlere Lötloch miteinander verbinden, aus. Jetzt haben Sie zwei Möglichkeiten: Wenn Sie nur noch mit EPROMs arbeiten wollen, könnten Sie die Widerstände wieder zwischen dem linken und dem mittleren Lötloch einsetzen. Die bessere Alternative ist es aber, dreipolige Pfostenstifte (sehen etwa so aus wie der Tastaturanschluß auf der Platine) einzulöten, und dann per Jumper (kleine Dinger, mit denen man zwischen jeweils zwei Pins Kontakt herstellen kann) zu wählen, ob ROMs (beide Jumper mitte-rechts) oder EPROMs (beide Jumper mitte-links) eingesteckt sind. Dieses Verfahren hat den Vorteil, daß relativ schnell die Chiptypen getauscht werden können.

Zur Sicherheit sollten Sie danach überprüfen, ob beim Löten Zinnspritzer auf die Platine gefallen sind, die Kurzschlüsse erzeugen könnten, und wenn ja, diese entfernen. Markieren Sie die ROMs (Aufkleber u.ä.), damit Sie später noch wissen, in welchem Sockel welcher ROM-Chip ge-



Anordnung der Lötbrücken

steckt hat, hebeln Sie sie dann mit einem kleinen Schraubenzieher aus den Sockeln und setzen Sie die EPROMs ein. Normalerweise zeigen die Kerben der Chips zu den Lötbrücken, und dort sind auch die vier freien Pins. Sollte das bei Ihrem Computer gerade andersherum sein, müssen Sie natürlich dann auch die EPROMs andersherum einstecken. Beim Einsetzen der EPROMs muß man beachten, daß das EPROM EE in den HI-Sockel (obere Fassung) und das EPROM E0 in den LO-Sockel (untere Fassung) gesteckt wird. Die Kerbe des EPROMs muß mit der des Sockels übereinstimmen, der Sockel ist jetzt komplett belegt.

Bauen Sie nun Ihren Computer für einen Testlauf, also noch nicht komplett, zusammen, schließen Sie nur Netzteil, Floppy, Tastatur und Monitor an. Die Platine sollte entweder auf Papier liegen oder wieder mit dem unteren Abschirmblech versehen werden (noch nicht festschrauben), damit der Untergrund keine Kurzschlüsse erzeugen kann. Schalten Sie den Monitor ein und legen Sie die Diskette ins Laufwerk. Der Computer sollte jetzt über eine Steckdosenleiste mit Schalter o.ä. ein- und ausgeschaltet werden (Gefahr am offenen Netzteil). Betätigen Sie nach dem Einschalten in kurzen Abständen die Leertaste. Wenn Sie den Tastenklick hören (Monitorlautstärke aufdrehen) und das Atarizeichen erscheint, ist erstmal alles in Ordnung. Der RAM-Test und die Warteschleife können mit einem Tastendruck abgebrochen werden, so daß der Computer sofort zu booten anfängt. Wenn er bis zum Desktop kommt, können Sie ihn wieder zusammenbauen.

Sollte sich aber nichts tun (Bildschirm bleibt dunkel, Laufwerk bootet nicht usw.), oder es erscheint eine Fehlermeldung, schalten Sie sofort alles aus, warten etwas (wegen der Spannung am Netzteil) und überprüfen Sie nochmal den Sitz aller Steckverbindungen, der EPROMs und eventuell die anderen gesockelten Chips auf der Platine (könnten locker sitzen, festdrücken). Sollte der Rechner danach immer noch nicht funktionieren, tauschen Sie die beiden EPROMs gegeneinander aus (das obengenannte Problem mit der Kennzeichnung) und versuchen es nochmal. Funktioniert er jetzt, können Sie ihn zusammenbauen, andernfalls testen Sie ihn mal mit TOS 1.06 (Brücken ändern, daher sind Pfostenstifte mit Jumpern vorteilhaft). Funktioniert er dann, sind vermutlich die TOS 2.06-EPROMs defekt.

Umbauhindernisse

Nur für den Fall, daß der 1040er einer Leserin/eines Lesers über ein anderes Platinen-Layout verfügt als der von mir umgebaute, will ich hier einige Hilfeansätze geben, damit der Computer doch noch umgerüstet werden kann. Alle Hindernisse sind aber theoretischer Natur, stammen

aus meinen Erfahrungen mit der ST-Reihe und müssen deshalb nicht unbedingt beim STE vorkommen. Falls Sie unlösbare Probleme haben, können Sie mit mir Kontakt aufnehmen (Telefon und Adresse weiter unten), ich will versuchen, Ihnen weiterzuhelfen.

Wenn TOS 1.06 auf 32poligen Chips vorhanden sein sollte, kann es sein, daß Sie einfach nur die Chips wechseln müssen. Dabei müssen Sie nur die Einsteckrichtung der Kerbe beachten und die entsprechenden Chips in die Sockel stecken (Chip E0 in LO, Chip EE in HI). Falls es nicht klappt, erstmal die Chips gegeneinander austauschen.

Sollten die Lötbrücken W102 und W104 nicht vorhanden sein, suchen Sie andere Lötbrücken auf der Platine. Falls Sie fündig werden, messen Sie mit einem Durchgangsprüfer, ob der jeweils mittlere Pin der Brücke mit Pin 2 oder Pin 24 der Sockel Verbindung hat. Wenn ja, haben Sie die entsprechenden Lötbrücken gefunden. Die Brücke mit Verbindung zu Pin 24 entspricht W102, die andere mit Verbindung zu Pin 2 W104.

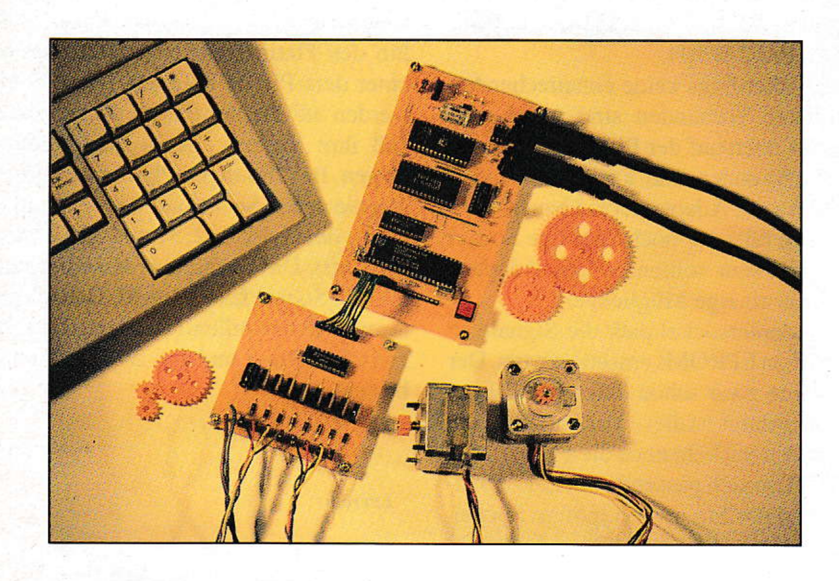
Wenn allerdings keine entsprechenden Lötbrücken vorhanden sind, wird man entweder doch auf der Platine Leiterbahnen durchtrennen und anders verbinden oder sich zwei Adaptersockel bauen müssen. Auch falls nur noch 28polige Sockel vorhanden sein sollten, wären Adaptersockel die einzige Möglichkeit.

Der Adaptersockel paßt die Signale so an, daß man EPROMs einsetzen kann. Der Bau wurde zwar schon einmal in [2] gezeigt, allerdings ist diese Variante zu hoch, da über den Sockeln noch die Floppy platziert ist. Um sich passende Sockel zu bauen, benötigen Sie zwei 32polige Sockel, an denen die Pins 1, 2, 24, 31 und 32 gekürzt werden. Pin 1, 32 und 31 werden mit Pin 30 (5 Volt) und Pin 24 mit Pin 16 (Masse) verbunden (daran denken, daß diese Sockel in die der Platine eingesteckt werden). Dann muß an Pin 2 ein Draht angelötet werden, den man später in den Pin des Platinensockels steckt, der sich unter dem Pin 24 befindet. Diese Sockel werden auf die der Platine so aufgesetzt, daß ihre Kerben alle in eine Richtung zeigen. Bei 28poligen Platinensockeln ragen die neuen Sockel auf der Seite mit der Kerbe über (also 1, 2, 31 und 32 hängen in der Luft). Die Drähte werden eingesteckt und die Sockel festgedrückt. Danach sollte man sicherstellen (durchmessen), daß zwischen Pin 2 und Pin 24 kein Kontakt besteht.

Harald Simon

Literatur:

[1] ST Computer 1/90, Patch As Patch Can
[2] ST Computer 6/90, TOS, TOS Hurra
[3] ST EXTRA 3, EPROM-Grundlagen



MIDI8031

Micro-Controller-Entwicklungssystem

Mikro-Controller steuern und regeln. Dazu müssen sie große Leistungen schalten können. Im letzten Teil des Projektes finden Sie daher eine, im wahrsten Sinne des Wortes, „kräftige" Schaltung: den MIDI8031-Power-Port. Für etwa 20 DM kann Ihr MIDI8031-Mikro-Controller-System damit bis zu 8 * 20 Ampère schalten! Als Anwendungsbeispiel dazu steuert ein GFA-BASIC-Programm mit Hilfe von MIDI8031 und dem Power-Port zwei Schrittmotoren.

Teil 3: Power-Port

Die Schaltung für den Power-Port ist prinzipiell recht einfach: Jedes der 8 Bits von Port 1 des Mikro-Controllers läuft über einen Inverter, dem jeweils ein SIPMOS-Leistungstransistor nachgeschaltet ist. SIPMOS-Transistoren haben nahezu ideale Schalteigenschaften. Sie lassen sich direkt mit den selben logischen Pegeln ansteuern, wie sie auf der MIDI8031-Platine vorkommen, also zwischen 0 und 5 Volt. Liegt der Steuereingang „G" auf logisch 0, sperrt der Transistor, bei logisch 1 wirkt er praktisch wie ein Kurzschluß! Der in der Schaltung verwendete BUZ10A kann bis zu 20 Ampère schalten. Parallel zur jeweiligen Last (im Schaltplan V1-V8) ist eine Diode geschaltet. Mit ihr hat es folgende Bewandtnis: Fließt ein Strom durch eine Spule (wie etwa ein Relais, entsteht ein Magnetfeld. Wird der Stromkreis mit einem Schalter unterbrochen, ist normalerweise ein kräftiger Funke zwischen den Kontakten des Schalters zu beobachten, da beim Zusammenbrechen des Magnetfeldes kurzzeitig hohe Spannungen genau entgegengesetzt zur vorher angelegten Spannung auftreten. Diese sogenannte Freilaufspannung würde den Transistor zerstören. Sie kann aber durch eine Diode kurzgeschlossen werden. In der Schaltung wurde der preiswerte Typ 1N4001 verwendet, der zwar kurzzeitig einige Ampère verträgt, aber bei sehr starken Verbrauchern mit Spulen (zum Beispiel großen Elektromotoren) durch einen geeigneteren Typ ersetzt werden sollte. Dabei sollten Sie auch beachten, daß die Kupferbahnen der Platine aufgrund ihrer Breite nur für eine Belastung von insgesamt etwa 5 Ampère ausgelegt sind. Die Schaltfrequenz des Power-Ports ist sehr hoch: sie liegt im Megahertz-Bereich! Aufgrund der Zykluszeit des 80C31 von einer Mikrosekunde lassen sich Lasten mit bis zu 500 Kilohertz schalten.

Eine interessante Nutzung des Power-Ports ist seine Verwendung als Schrittmotorsteuerung. Sogenannte „unipolare" Schrittmotoren, wie sie sehr häufig angeboten werden, sind besonders geeignet für den Power-Port. Das Funktionsprinzip eines unipolaren Schrittmotors ist in Bild 4 skizziert. Tatsächlich ist in Wirklichkeit die Winkelauflösung eines Schrittmotors viel höher als in Bild 4, aber darauf soll es hier nicht ankommen, zumal sich am System der Reihenfolge der Spulenströme nichts ändert. Der Widerstand in Bild 4 hat den Zweck, das Drehmoment des Motors bei hohen Drehzahlen zu verbessern: je höher die Spulenspannung ist, desto schneller wird das Magnetfeld aufgebaut. Der Widerstand begrenzt den Maximalstrom auf zulässige Werte. Der Wert des Widerstandes liegt in der Regel im Bereich des ohmschen Widerstandes einer der 4 Spulenhälften. Da zur Steuerung eines Schrittmotors nur 4 Schalter benötigt werden, lassen sich 2 Schrittmotoren an den Power-Port anschließen!

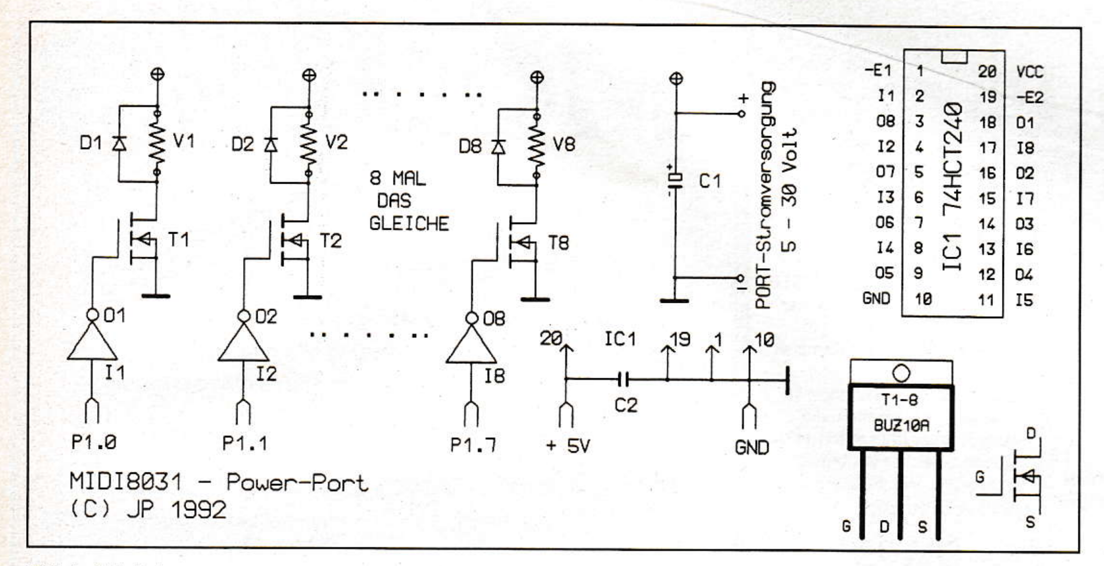


Bild 1: Schaltplan

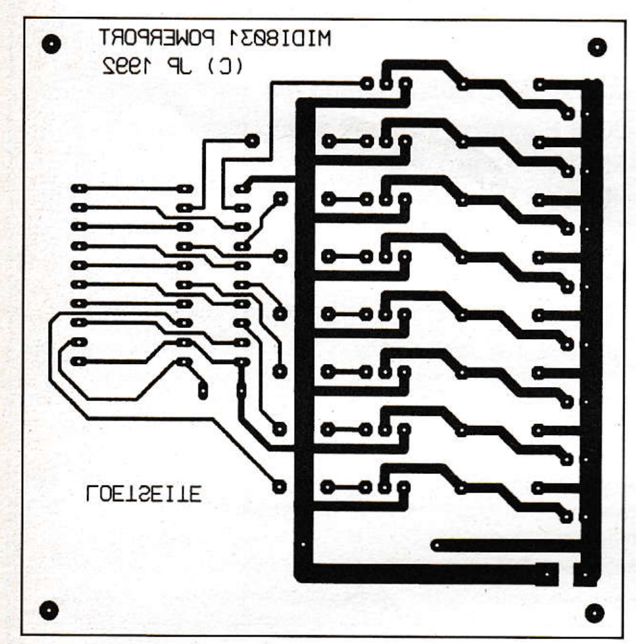


Bild 2: Platine

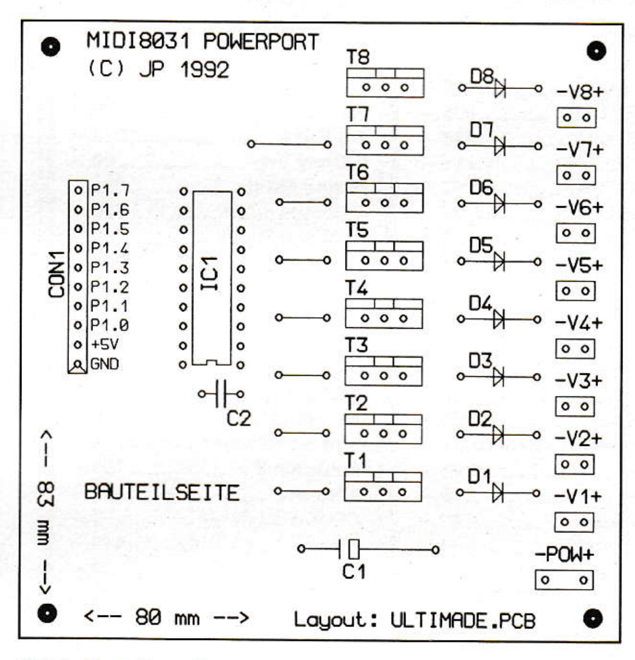


Bild 3: Bestückungsliste

Der Mikro-Controller 80C31 kennt 5 verschiedene Interrupt-Quellen. Zwei Interrupts können jeweils über die beiden Timer T0 und T1 ausgelöst werden. Um die Programmierung von Interrupts auf dem MIDI8031-Mikro-Controller-Board zu demonstrieren, schaut der Timer0-Interrupt 3900 mal pro Sekunde nach, ob einer der beiden Motoren einen Schritt weitergeschaltet werden soll. Alle Variablen werden im internen RAM des 80C31 abgelegt. Für jeden Motor gibt es die Variablen *speed0/1* und *nach0/1*. Bei jedem Interrupt springt der 80C31 zur Routine „steppen". Für jeden Motor addiert er dann dessen Variablen *speed* und *nach* und speichert das Ergebnis in *nach* ab. Überschreitet das Ergebnis die 8-Bit-Grenze, wird das Carry-Flag gesetzt und der betreffende Motor um eine Position vorwärts oder rückwärts bewegt, je nachdem ob das Bit 7 der Variablen *speed* gesetzt ist oder nicht. Das Hauptprogramm ab *go* initialisiert den Interrupt und die Variablen. Zu Anfang drehen sich beide Motoren mit minimaler Geschwindigkeit. Das Programm inaktiviert den seriellen Interrupt des 80C31, über den dieser mit dem Entwicklungssystem auf dem ATARI kommuniziert, und verteilt ankommende MIDI-Bytes abwechselnd an die Geschwindigkeits- variablen der beiden Motoren. Daher sollten Sie das Programm von der Shell des Entwicklungssystems aus mit der Taste „X" starten und die Shell danach verlassen, ins GFA-BASIC wechseln und das BASIC-Programm laden und starten. Sofort wird sich emsige Aktivität unter Ihren Motoren ausbreiten: vorwärts, rückwärts, schneller, langsamer ... Die Motordrehzahl wird auf dem Bildschirm dargestellt.

So, damit wären wir am Ende des ST-Computer-Mikro-Controller-Projekts. Sicher haben Sie inzwischen schon jede Menge eigene Ideen, die nur darauf drängen, verwirklicht zu werden. MIDI8031 und ATARI sind ein prima Gespann. Wir würden gerne über Ihre Projekte berichten!

Jürgen Piscol

Stückliste MIDI8031-POWERPORT

IC1:	74HCT240 od. 74HC240 od.74LS240,8-Inverter, DIL2
T1-T8:	BUZ10A SIPMOS, Leistungs-MOS-FET50V/22A
D1-D8:	1N4001 Diode 50V/1A evtl.stärkerer Typ
C1:	100µF/50V, axial
C2:	100nF, Rastermaß 5mm
CON1:	Buchsenleiste 10polig

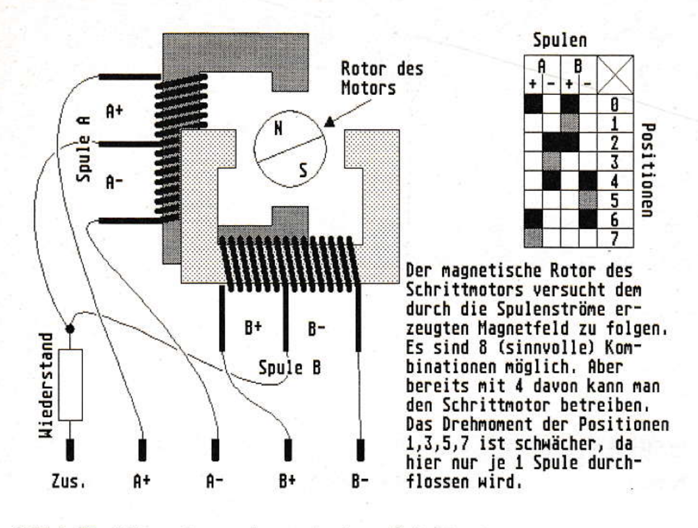


Bild 4: Funktionsschema eines unipolaren Schrittmotors

```
;
; * SCHRITT: Dieses Programm steuert 2 Schritt-
; * motoren. Die Drehzahl der beiden Motoren
; * empfaengt MIDI8031 ueber MIDI (was sonst),
; * dazu dient das Steuerprogramm in GFA-Basic
; * (c)1992 by MAXON-Computer GmbH
; * Autor: Jürgen Piscol
;

    .registers reg51.inc    ; 8031-Deklarationen

; *** Variablen Internes RAM, oberhalb Bitfeld
    .var_org $30        ; Variablen-PC setzen

speed0: .ds.b 1         ; -64..0..64  Gessch. M. 0
nach0:   .ds.b 1        ; Nachkommastellen
pos0: .ds.b 1           ; Magnetfeldpos. (0..3)
speed1:   .ds.b 1       ; dto. Motor 1
nach1:    .ds.b 1       ; dto.
pos1: .ds.b 1           ; dto.
awert:    .ds.b 1       ; Hilfsvariable
stack:                  ; Ab hier nur noch Stack

; *** Ab hier beginnt der Programmcode ***

    .text_org $8000

; *** 'Interruptbereich' wie EPROM ab Adr. $0 ***
        ljmp go        ; Int.ber.. ueberspringen
int0: reti             ; Nicht verwendet...
        .ds.b 7
timer0:  ljmp steppen    ; zum Steppen springen
        .ds.b 5
int1: reti             ; int0/1, timer1, serial
        .ds.b 7         ; werden in diesem Prog.
timer1:  reti           ; nicht verwendet. Der
        .ds.b 7         ; Programmkopf zeigt
serial:  reti           ; lediglich, welche Inter.
        .ds.b 7         ; MIDI8031 alle kennt!

; *** Hier startet das eigentliche Programm ***

; * Initialisierungen: Variablen & Interrupts
go:
    mov sp,#stack       ; !!! Stack einrichten !!!

    setb TCON.4         ; TR0 Timer 0 laeuft
    anl TMOD,#%11111100  ; Mode-Bits ausblenden
    orl TMOD,#%11       ; M0=3: Int.frq. 3.9KHz
    setb IE.7           ; Gen. Interrupts zulassen
    setb IE.1           ; T0 Interrupt Ein
    setb IP.1           ; T0 Int. Hohe Prioritaet

    mov speed0,#1       ; Motor1: (langs. vorw.)
    mov nach0,#0        ;         Nachkommastellen
    mov pos0,#0         ;         Motor1 an Pos. 0!
    mov speed1,#-1      ; Motor2: (langs. rueckw.)
    mov nach1,#0        ;
    mov pos1,#0         ;

; * Warteschleife: Abwechselnd werden MIDI-Bytes
; * an speed0/1 verteilt.

    clr IE.4           ; Seriellen Int. verbieten
    clr SCON.0         ; Ser. Buffer loeschen
loop:
    cpl P3.5          ; Anzeigen ueber LED
    jnb SCON.0,!        ; RI=1 abwarten (1: Byte
    clr SCON.0          ; angekommen), quittieren
    mov speed0,SBUF       ; Byte an speed0
    jnb SCON.0,!
    clr SCON.0
    mov speed1,SBUF       ; Byte an speed1
    sjmp loop

; *** Stepper-Interr., Aufruf mit 3.9kHz ***
steppen:
    push ACC          ; Akku retten
    push PSW          ; Flags retten
; * Step-Routine fuer Motor 0 *
    mov A,nach0         ; Addieren
    add A,speed0        ; des Steps
    mov nach0,A         ; und Abspeichern...
    jnb PSW.2,nstep0    ; Overflow-Flag testen
    mov awert,#1        ; Hilfsvar: Vorwaerts
    jb ACC.7,vor0       ; >$80: Vorwaerts ist ok
    mov awert,#-1       ; sonst Hilfsv = Rueckw.!
vor0:
    mov A,pos0          ; Position 0 holen
    add A,awert         ; 1 Step vor/zurueck
    anl A,#%0011        ; 4 Steps fuer Motor 0
    mov pos0,A          ; Neue Position merken
nstep0:

; * Step-Routine fuer Motor 1 *
    mov A,nach1         ; wie Motor 0
    add A,speed1        ; ...
    mov nach1,A
    jnb PSW.2,nstep1
    mov awert,#4        ; Vorw. in 4-er Schritten
    jb ACC.7,vor1
    mov awert,#-4       ; sonst 4 Rueckwaerts!
vor1:
    mov A,pos1          ; Position 1 holen
    add A,awert
    anl A,#%1100        ; 4 Steps f. M1 (0,4,8,12)
    mov pos1,A
nstep1:

; * Ausgaenge schalten in Abh. von pos0/1 *
    mov A,pos0          ; 0..3 Step 1
    add A,pos1          ; 0..15 Step 4
    add   A,#pot-h1       ; Adressdifferez zur Tab.
    movc A,@A+pc        ; Indiziert aus Tabelle
h1:   mov P1,A          ; Bitmuster am Port ausg.

    pop PSW             ; Flags restaurieren
    pop ACC             ; Akku restaurieren
    reti              ; Interrupt fertig!

; * Positionstabelle fuer 16 Strommoeglichkeiten *
pot:
    .dc.b $AA,$A9,$A5,$A6,$9A,$99,$95,$96
    .dc.b $5A,$59,$55,$56,$6A,$69,$65,$66

```

Listing 1

```
' *** MIDI8031 Mikrocontrollerboard ***
' Nutzung der MIDI8031-Schrittmotorsteuerung
' ueber GFA-Basic. Drehzahl jedes Motors von
' -100%..0%..+100% separat einstellbar!
'
'(c)1992 by MAXON-Computer GmbH
'Autor: Jürgen Piscol
'
px=10      ! Plotposition X-Achse
max=32     ! Max.geschw. der Mot.
DIM soll(1),ist(1)  ! Je 2 SOLL/IST-Werte
PRINT „Ende: <Taste>"
REPEAT
  FOR i=0 TO 1             ! Motoren 0 u. 1
    IF RND(0)<0.01         ! evtl. SOLL neu
      soll(i)=RANDOM(max*4+1)-max*2 ! -max..+max
    ENDIF                           ! IST folgt
    ADD ist(i),SGN(soll(i)-ist(i))  ! langsam
    PLOT px,100+ist(i)+i*200        ! SOLL.
    OUT 3,ist(i)/2                  ! SOLL send.
  NEXT i                          ! Fuer jeden Mot.
  PAUSE 1                  ! Kurze Pause
  px=px+0.1                ! Plotpos. etwas weiter
UNTIL INKEY$<>""           ! <Taste>: Ende
OUT 3,0                    ! Alle Maschinen STOP!
OUT 3,0
END
```

Listing 2

Kugelsicher

Resetfeste Programme auf dem ST
Der Nachtrag

Tja, durch die Aussicht auf den baldigen Urlaub hat es uns in der letzten Ausgabe ein wenig an Konzentration gemangelt. Die Listings zu unserem Artikel "Kugelsicher - resetfeste Programme" waren falsch bzw. nicht vorhanden. Aus diesem Grund streuen wir uns hiermit Asche aufs Haupt, ziehen den Sackleinen an und reichen die Listings selbstverständlich nach. Nostra culpa, nostra maxima culpa!

```
; ###############################################
; ##   *** KUCKUCK! ***                        ##
; ##   RESETFESTES DEMONSTRATIONSPROGRAMM      ##
; ##   by ROLAND NITZ  (c) 1992 MAXON Computer ##
; ###############################################

        .TEXT
START:  bra     MAIN        ;
; » WEGEN RESIDENT-BLEIBEN... «

; ** WIRD PER RESETVEKTOR ANGESPRUNGEN...***+
; KOPIERT "RESET-ROUTINE" AN EINGETRAGENE ADRESSE
; (SIEHE OBEN), SETZT PHYSTOP HERAB UND KLINKT
; SICH DANN AUS EINER EVT. XBRA-KETTE WIEDER AUS.

        .DC.b "XBRA"        ; XBRA-PROTOKOLL
        .DC.b "KUCK"        ; PROGRAMMKENNUNG
OLDRV:  .DC.l 0 ;           ; ALTE ADRESSE
RESVEC: movem.l d0-d2/a0-a2/a7,SAVREGS
        lea.l   MAIN+1052,sp ; STACK INIT.
        movea.l $42e,a2     ; PHYSTOP HOLEN
        suba.w  #((ERT-SRT+$1ff)&$200),a2
;                           ; VERSCHIEBUNG
        lea.l   SRT,a0      ; STARTADRESSE
        movea.l a2,a1       ; COPY-ADRESSE
        move.l  a1,$42e     ; NEUES PHYSTOP
        move.w  #(ERT-SRT)/4,d0
.COPY:  move.l  (a0)+,(a1)+ ; KOPIERE DATEN
        dbra    d0,.COPY    ; NÄCHSTEN...

; » ADRESSE IM HANDLER EINTRAGEN
        lea.l   $680,a0     ; SPRUNGLISTE
        moveq.l #32-1,d0    ; MAX. 32 POS.
.TSTPOS: tst.l  (a0)+       ; IST 0?
        beq     .OKPOS      ; JA: OK!
        dbra    d0,.TSTPOS  ; WEITERSUCHEN

        tst.w   d0          ; D0 NEGATIV?
        bmi     .FAILIT     ; JA: AUS.

.OKPOS: move.l  a2,-4(a0)   ; ROUT. EINHÄNGEN
        bsr     CHCKSM      ; PRÜFSUMME KORR.

.FAILIT: lea.l  $42a,a1     ; BASISADRESSE
.TST:   cmpi.l  #RESVEC,(a1) ; BIN DAS ICH?
        beq     .CPY        ; JA: KOPIERE...
        movea.l (a1),a2     ; ADR. HOLEN
        cmpi.l  #"XBRA",-12(a2) ; XBRA?
        bne     .REMOVE     ; NEIN: DANN CLR!!
        lea.l   -4(a2),a1   ; FÜR NÄCHSTEN LAUF
        tst.l   (a1)        ; ADRESSE 0?
        bne     .TST        ; NEIN: WEITER
        bra     .EXIT       ; JA: RAUS (??!)
.CPY:   move.l  OLDRV,(a1)  ; SONST SETZEN

        tst.l   $42a        ; RES_VECTOR=0?
        bne     .EXIT       ; NEIN: WEITER

.REMOVE: clr.l  $426        ; RESVALID=INVALID

.EXIT:  tst.l   OLDRV       ; NOCH EINE ROUTINE?
        beq     .CONT       ; NEIN: WEITER
        movem.l SAVREGS,d0-d2/a0-a2
        move.l  OLDRV,-(sp) ; ALTE ROUTINE
        rts                 ; UND ZURÜCK

.CONT:  cmpi.w  #$9bcd,(a6) ; FEHLERHAFTES TOS?
        bne     .GOON       ; NEIN: NORMAL WEITER
        lea.l   $24(a6),a6  ; SONST $24 ADDIEREN
.GOON:  movem.l SAVREGS,d0-d2/a0-a2/a7
;                           ; REGISTER RETTEN
        jmp     (a6)        ; UND WEITERMACHEN

;*** HANDLER ***
SHANDLER:
        .DC.l $12123456     ; MAGIC NUMBER
        .DC.l $600          ; ZEIGER
        movem.l d0-d7/a0-a6,-(sp)
        movea.w #$680,a0    ; STARTADRESSE
        move.w  #16-1,d0    ; MAXIMAL 16 ROUTINEN
.EXEC:  tst.l   (a0)+       ; IST ADRESSE=0?
        beq     .CONT       ; JA: WEITER
        movea.l -4(a0),a1   ; SONST ADRESSE HOLEN
        movem.l d0/a0,-(sp) ; SAVE REGISTER
        jsr     (a1)        ; UND AUSFÜHREN
        movem.l (sp)+,d0/a0 ; RESTORE REG.
.CONT:  dbra    d0,.EXEC    ; NÄCHSTE ADR
        movem.l (sp)+,d0-d7/a0-a6
        rts                 ; ZURÜCK
EHANDLER:

; *** EIGENTLICHE ROUTINE, UM DIE ES GEHT... ***
SRT:    bra.s   .JMP        ; ÜBERSPRINGEN
        .DC.l "KUCK"        ; KENNUNG...
.JMP:   pea     RESTXT(pc)  ; TEXT
        move.w  #9,-(sp)    ; F#9 : CCONWS
        trap    #1          ; GEMDOS
        addq.l  #6,sp       ; STACKKORREKTUR
W_KEY:  move.w  #7,-(sp)    ; FUNKTION #7
        trap    #1          ; GEMDOS: CRAWCIN
        addq.l  #2,sp       ; STACKKORREKTUR
        rts                 ; ZURÜCK
RESTXT: .DC.b 27,"p KUCKUCK! "
        .DC.b 27,"q [TASTE]",13,10,0
        .EVEN
ERT:                        ; ENDE DER ROUTINE

; *** ROUTINE KORRIGIERT PRÜFSUMME ***
CHCKSM: moveq.l #0,d0       ; D0 LÖSCHEN
        movea.w #$600,a0    ; SEITE LADEN
        move.w  #256-2,d1   ; 255 WÖRTER TESTEN
.ADD:   add.w   (a0)+,d0    ; UND ADDIEREN
        dbra    d1,.ADD     ; NÄCHSTES WORT

        move.w  #$5678,d2   ; PRÜFSUMME
        sub.w   d0,d2       ; AKT. PRÜFSUMME -
        move.w  d2,(a0)+    ; NEUE EINTRAGEN

        rts                 ; UND TSCHÜSS..

;*** DIESER TEIL BLEIBT NICHT RESIDENT! ***
SAVREGS:
```
→

```
MAIN:     clr.l    -(sp)        ; MODUS: SV
          move.w   #32,-(sp)    ; F# 32
          trap     #1           ; GEMDOS: SUPER
          addq.l   #6,sp        ; STACKKORREKTUR
          move.l   d0,OLDSP     ; ALTEN SSP MERKEN

          lea.l    $600,a0      ; BASISADRESSE
          cmpi.l   #$12123456,(a0) ; SCHON INST.?
          bne      .INST        ; NEIN: INSTALLIEREN
          cmpa.l   4(a0),a0     ; STIMMT DAS AUCH?
          bne      .INST        ; NEIN: INSTALLIEREN
          moveq.l  #0,d1        ; LÖSCHEN (PRÜFSUMME)
          move.w   #$100-1,d0   ; 256 WÖRTER ADDIEREN
.TSTCHS:  add.w    (a0)+,d1     ; ADDIEREN
          dbra     d0,.TSTCHS   ; NÄCHSTE
          cmpi.w   #$5678,d1    ; PRÜFSUMME KORREKT?
          beq      .JUMP        ; JA: WEITER

.INST:    bsr      H_INST       ; < EIGENER HANDLER >
          bsr      CHCKSM       ; CHECKSUM-KORR.
          bra      .NOTFND      ; KANN NOCH NICHT
;                               ; INSTALLIERT SEIN

.JUMP:                 ; TESTET, OB PRG SCHON RESIDENT
          lea.l    $680,a0      ; BASISADRESSE
          moveq.l  #32-1,d0     ; 32 SLOTS TESTEN
.TST:     tst.l    (a0)+        ; WERT=0?
          beq      .CONT1       ; ÜBERGEHEN
          movea.l  -4(a0),a1    ; SONST ADR HOLEN
          cmpi.l   #"KUCK",2(a1) ; STIMMT KENNUNG?
          beq      .FOUND       ; JA: GEFUNDEN
.CONT1:   dbra     d0,.TST      ; SONST: WEITERSUCHEN

          cmpi.l   #$31415926,$426 ; RESVALID?
          bne      .NOTFND      ; NICHT GEFUNDEN!
          movea.l  $42a,a0      ; SONST ADR HOLEN
.TST2:    cmpi.l   #"XBRA",-12(a0) ; XBRA?
          bne      .NOTFND      ; NEIN: >RAUS
          cmpi.l   #"KUCK",-8(a0) ; KENNUNG ?
          beq      .FOUND       ; STIMMT: GEFUNDEN
          tst.l    -4(a0)       ; IST ADR =0?
          beq      .NOTFND      ; => LETZTE ROUTINE
          movea.l  -4(a0),a0    ; SONST ADR HOLEN
          bra      .TST2        ; UND WEITERSUCHEN

.NOTFND:  cmpi.l   #$31415926,$426 ; RESVALID?
          bne      .NORM        ; NEIN: NORMAL WEITER
          move.l   $42a,OLDRV   ; SONST: ADR. MERKEN
.NORM:    move.l   #RESVEC,$42a ; NEUE ROUT. INST.
          move.l   #$31415926,$426 ; RESVALID!

.K_RES:   pea      R_OK         ; TXT:"PRG INST."
          move.w   #9,-(sp)     ; F#=9
          trap     #1           ; GEMDOS: CCONWS
          addq.l   #6,sp        ; STACKKORREKTUR
          bsr      W_KEY        ; WARTE AUF TASTE
          bsr      SUPOFF       ; SV-MODUS AUS
          clr.w    -(sp)        ; STATUS: 0=OK
          move.l   #MAIN-START+1308,-(sp)
;         SOVIELE BYTES RESIDENT HALTEN...
          move.w   #49,-(sp)    ; F# 49
          trap     #1           ; GEMDOS

.FOUND:   pea      R_NI         ; TXT:"SCHON INST."
          move.w   #9,-(sp)     ; F#=9
          trap     #1           ; GEMDOS: CCONWS
          addq.l   #6,sp        ; STACKKORREKTUR
          bsr      W_KEY        ; WARTE AUF TASTE
          bsr      SUPOFF       ; SV-MODUS AUS
          clr.w    -(sp)        ; F#0
          trap     #1           ; GEMDOS: PTERM

; *** SUPERVISOR-MODUS AUS ***
SUPOFF:   movea.l  (sp)+,a3     ; RÜCKSPRUNGADR.
          move.l   OLDSP,-(sp)  ; ALTER STACK
          move.w   #32,-(sp)    ; F# 32
          trap     #1           ; GEMDOS: SUPER
          addq.l   #6,sp        ; STACKKORREKTUR
          jmp      (a3)         ; UND ZURÜCK...

; *** INSTALLIERT HANDLER ***
H_INST:   move.w   #(EHANDLER-SHANDLER)/4,d0
;                               ; LÄNGE IN LONGS
          movea.w  #$600,a1     ; KOPIERZIEL
          lea.l    SHANDLER,a0  ; STARTADR. QUELLE
.COPY1:   move.l   (a0)+,(a1)+  ; KOPIERE...
          dbra     d0,.COPY1    ; SOVIELE LANGWORTE

.CLR:     cmpa.l   #$800,a1     ; SCHON BEI  $800?
          beq      .EXIT        ; JA: EXIT
          clr.l    (a1)+        ; SONST LÖSCHEN
          bra      .CLR         ; UND NOCH EINMAL
.EXIT:    rts                   ; CIAO...

          .DATA
R_OK:     .DC.b "E KUCKUCK! ist jetzt"
          .DC.b " resident.",13,10,0
R_NI:     .DC.b "E KUCKUCK! war bereits"
          .DC.b " installiert.",13,10,0
          .EVEN

          .BSS
OLDSP:    .DS.l 1
```

```
; ################################################
; ##    *** R_KUCK! ***                         ##
; ##  ENTFERNT KUCKUCK AUS DEM SPEICHER          ##
; ##  by ROLAND NITZ  (c) 1992 MAXON Computer    ##
; ################################################
          .ABS
KUCKLEN   equ $200

          .TEXT
START:    clr.l    -(sp)        ; MODUS: SV
          move.w   #32,-(sp)    ; F# 32
          trap     #1           ; GEMDOS: SUPER
          addq.l   #6,sp        ; STACKKORREKTUR
          move.l   d0,OLDSP     ; ALTEN SSP MERKEN

          lea.l    $600,a0      ; BASISADRESSE
          cmpi.l   #$12123456,(a0) ; SCHON INST.?
          bne      .N_INST      ; NEIN: INSTALLIEREN
          cmpa.l   4(a0),a0     ; STIMMT DAS AUCH?
          bne      .N_INST      ; NEIN: INSTALLIEREN
          moveq.l  #0,d1        ; LÖSCHEN (PRÜFSUMME)
          move.w   #$100-1,d0   ; 256 WÖRTER ADDIEREN
.TSTCHS:  add.w    (a0)+,d1     ; ADDIEREN
          dbra     d0,.TSTCHS   ; NÄCHSTE
          cmpi.w   #$5678,d1    ; PRÜFSUMME KORREKT?
          bne      .N_INST      ; JA: WEITER

          lea.l    $680,a0      ; BASISADRESSE
          moveq.l  #32-1,d0     ; 32 SLOTS TESTEN
.TST:     tst.l    (a0)+        ; WERT=0?
          beq      .CONT1       ; ÜBERGEHEN
          movea.l  -4(a0),a1    ; SONST ADR HOLEN
          cmpi.l   #"KUCK",2(a1) ; STIMMT KENNUNG?
          beq      .FOUND       ; JA: GEFUNDEN
.CONT1:   dbra     d0,.TST      ; SONST: WEITERSUCHEN

          cmpi.l   #$31415926,$426 ; RESVALID?
          bne      .N_INST      ; NICHT GEFUNDEN!
          lea.l    $42a,a1      ; SONST BASIS HOLEN
.TST2:    movea.l  (a1),a0      ; ADR HOLEN
          cmpi.l   #"XBRA",-12(a0) ; XBRA?
          bne      .N_INST      ; NEIN: >RAUS
          cmpi.l   #"KUCK",-8(a0) ; KENNUNG ?
          beq      .RVFOUND     ; STIMMT: GEFUNDEN
          tst.l    -4(a0)       ; IST ADR =0?
          beq      .N_INST      ; => LETZTE ROUTINE
          lea.l    -4(a0),a1    ; SONST ADR HOLEN
          bra      .TST2        ; UND WEITERSUCHEN

.N_INST:  pea      NTEXT        ; TXT:"WAR N. INST."
          bra      .PRGEX       ; ->RAUS

.RVFOUND:
          move.l   -4(a0),(a1)  ; IM RESVEC GEFUNDEN
          tst.l    $42a         ; RES_VEC=0?
          bne      .REM         ; NEIN: WEITER
          clr.l    $426         ; SONST RESVALID DEL.
          bra      .REM         ; LÖSCHEN UND WEITER

.FOUND:   lea.l    -4(a0),a1    ; ZEIGER AUF ADRESSE
          lea.l    $680,a0      ; BASIS HOLEN
          movea.l  (a1),a2      ; ADRESSE HOLEN
          movea.l  $42e,a3      ; PHYSTOP HOLEN

          movea.w  #-1,a4       ; ADR.
          moveq.l  #32-1,d0     ; 32 SLOTS TESTEN
.GMIN:    tst.l    (a0)+        ; IST 0?
          beq      .CONT        ; JA: WEITERSUCHEN
          cmpa.l   -4(a0),a4    ; KLEINER ALS ADR?
          bls      .JMP         ; JA: WEITER
```

→

```
             movea.l -4(a0),a4  ; SONST HOLEN
    .JMP:    cmpa.l  -4(a0),a2  ; KLEINER ALS
             bls     .CONT      ; A2? JA: WEITER
             addi.l  #KUCKLEN,-4(a0) ;  ADR. KORR.
    .CONT:   dbra    d0,.GMIN   ; WEITERSUCHEN

             clr.l   (a1)       ; EINTRAG LÖSCHEN

             cmpa.l  a4,a3      ; KLEINSTE ADR
             bne     .REMERR    ; MUß=PHYSTOP SEIN

             move.l  a2,d0      ; ADRESSDIFFERENZ
             sub.l   a4,d0      ; ERRECHNEN.
             beq     .CNT2      ; IST =0? ->RAUS

             subq.l  #1,d0      ; WG. DBRA
    .SHIFT:  move.b  -(a2),KUCKLEN(a2) ; SCHIEBE...
             dbra    d0,.SHIFT  ; UND WEITER...

    .CNT2:   moveq.l #0,d0      ; D0 LÖSCHEN
             movea.w #$600,a0   ; SEITE LADEN
             move.w  #256-2,d1  ; 255 WÖRTER TESTEN
    .ADD:    add.w   (a0)+,d0   ; UND ADDIEREN
             dbra    d1,.ADD    ; NÄCHSTES WORT

             move.w  #$5678,d2  ; PRÜFSUMME
             sub.w   d0,d2      ; AKT. PRÜFSUMME -
             move.w  d2,(a0)+   ; NEUE EINTRAGEN

             addi.l  #KUCKLEN,$42e ; PHYSTOP KORR.
             bra     .REM       ; EXIT ERFOLGSMELDUNG

    .REMERR: pea     FEHLER     ; "FEHLER!!"
             bra     .PRGEX     ; ->RAUS

    .REM:    pea     REMOVED    ; TXT:"PRG. ENTFERNT"
    .PRGEX:  move.w  #9,-(sp)   ; F#:9 CCONWS
             trap    #1         ; GEMDOS-CALL
             addq.l  #6,sp      ; STACKKORREKTUR

             move.w  #7,-(sp)   ; #7: CRAWCIN
             trap    #1         ; (GEMDOS)
             addq.l  #2,sp      ; STACKKORREKTUR

             bsr     SUPOFF     ; SV-MODUS AUS

             clr.w   -(sp)      ; F#0: PTERM
             trap    #1         ; (GEMDOS)

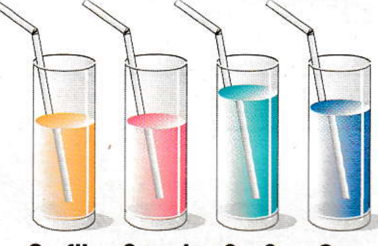
    ; *** SUPERVISOR-MODUS AUS ***
    SUPOFF:  movea.l (sp)+,a3   ; RÜCKSPRUNGADR.
             move.l  OLDSP,-(sp) ; ALTER STACK
             move.w  #32,-(sp)  ; F# 32
             trap    #1         ; GEMDOS: SUPER
             addq.l  #6,sp      ; STACKKORREKTUR
             jmp     (a3)       ; UND ZURÜCK...

             .DATA
    NTEXT:   .DC.b "E KUCKUCK! konnte nicht"
             .DC.b " gefunden werden.",13,10,0
    FEHLER:  .DC.b "E KUCKUCK! nur desaktiviert."
             .DC.b " FEHLER: Speicher ungenutzt!! ",0
    REMOVED: .DC.b "E KUCKUCK! wurde aus"
             .DC.b " dem Speicher entfernt.",13,10,0
             .EVEN

             .BSS
    OLDSP:   .DS.l 1
```







Grafik Sound Spaß Ges.

Push Over

Hersteller: Ocean
Vertrieb: Bomico

Wer sich mal herzhaft in einen Ameisenhaufen gesetzt hat, weiß, daß die kleinen Krabbelbiester echte Kämpfernaturen sind. Mit einem eher sympatischen Vertreter dieser Tierart wird der Spieler in Oceans neuem Geschicklichkeitsgame „Push Over" bekannt.

Held ist „G.I. Ant", eine sehr menschenfreundliche Ameise. Sie hift dem verzweifelten - und etwas dösigen - Colin. Seine Quaver-Kornflakes sind dummerweise in einen Ameisenhügel gepurzelt. Über den Geschmack des Flockenfutters ließe sich sicher streiten - aber, einerlei, G.I.Ant macht sich sofort auf die Suche nach den gräßlichen Quavers. Zur allgemeinen Überraschung muß er sich bei der Flockensuche mit den phantasievollsten Dominovarianten herumärgern - oder amüsieren, wie man's nimmt. Vor prächtigen Hintergrundillustrationen erstrecken sich etliche Plattformen auf dem Screen. Recht unordentlich stehen überall große Dominosteine herum. G.I.Ant muß die Spielklötze nun herumschleppen und in einer bestimten Weise aufbauen. Bekanntlich können Ameisen ja ein Vielfaches ihres Körpergewichtes tragen - kein Problem also für Super-Ant. Der arme Spieler denkt für das starke Tier und dirigiert es mit Joystick oder Tasten 100 Levels lang. Alle 10 Levels verwöhnt das Programm seinen Fan mit einer Zwischengrafik: Colin bekommt darin seine Quavers. Jeweils daran anschließend, werkelt die tolle Ameise vor einem neuen Hintergrund. Mal qualmen Fabriken hinten im Screen, später versetzen die Bilder in die Antike, ins Reich der Azteken und in eine Weltraumstation. Fünf Ameisenleben liegen auf der digitalen Reservebank. Die kann man brauchen, denn wenn G.I.Ant mal von einer hohen Plattform klatscht, ist das Tierchen hinüber. Manchmal muß man dieses Risiko eingehen, da alles sehr, sehr schnell gehen muß. In der Eile klettert G.I. Ant dann nicht betulich die Leitern runter, sondern macht einen kühnen Satz. Wenn er dabei ein klein bißchen auf die Nase fällt, fühlt er sich allenfalls etwas duselig, kann aber weitermachen. Level für Level werden die neun Dominovarianten kniffeliger. In leichteren Spielstufen geht es zunächst nur darum, die schwarzweißen Steine so aufzureihen, daß der gestreifte Endklotz auch ganz bestimmt umfällt, wenn das erste Steinchen angestupst worden ist. Oceans Spielemacher haben sich dazu die verrücktesten Sachen ausgedacht. Zum Beispiel werden die Dominosteine auch dazu benutzt, von einer Plattform zur anderen zu gelangen. Einmal haut der Endstein ein Loch in die Plattform, so daß G.I. Ant eine Etage tiefer springen kann. Klaft zwischen zwei Ebenen eine tiefe Schlucht, hat auch da das Dominoverfahren eine Lösung: der letzte Stein muß so umfallen, daß er wie eine Zugbrücke über die Kluft fällt. Und dann wird es kompliziert. Es gibt nämlich eine Art Anstoßsteine, die erst einmal ins Rollen gebracht werden, an einem roten Stopper abprallen und dann plangemäß die nächste Reihe Dominos umhauen. Ei freilich gibt es auch fliegende Dominosteine, die einander schwebend anditschen.

Glücklichen Absolventen vorheriger Levels gibt ein Ameisenheer die Zugangsnummer zum nächsten Teil des Vergnügen heraus und legt noch einen Gutschein bei. Mit einem solchen sogenannten Token darf man das verhunzte Level erneut angehen. Ein Glück - denn manchmal sind die Kombinationsmöglichkeiten so groß, daß man selbst mit heißgelaufenem Grips nicht so leicht auf die Lösung kommt. Wie schön wäre es in diesem Fall, zu zweit an dem Problem zu arbeiten. Aber einen Zwei-Spieler-Modus hat Ocean uns denn doch nicht gegönnt. Dabei ist das Spiel mit den labberigen Kornflakes und dem freundlichen Piekstier ansonsten ein echter Knaller. Die Bilder sind sehr schön, der heiße Jazz im Hintergrund klasse. Auch die Steuerung ist akkurat und schnell. Ein besonderes Lob noch für die Bildermaler: die Figur von G.I. Ant ist so süß wie seinerzeit die schnuckeligen kleinen Lemminge. Mit der Sprache gibt es keine Schwierigkeiten: zuvorkommend wird das Game in vier Sprachen ausgeliefert. Jaja, auch in Deutsch. Englandreisende werden eine Werbung für dieses Spiel übrigens auch auf den Quavers-Packungen finden. Ocean und der Flockenhersteller haben nämlich einen Deal gemacht: Quavers kommen im Spiel vor, andersrum wird das Spiel auf der Cornflakes-Packung beworben. Nun sollte man daraus aber nicht schließen, daß Spiel und Knusperfutter die gleiche Qualität haben. Quavers haben dem - nicht gerade verwöhnten - Tester so gut geschmeckt wie der Sandkuchen damals in der Buddelkiste.

Carsten Borgmeier

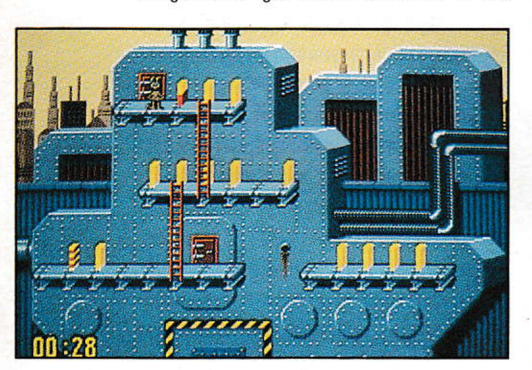
Epic

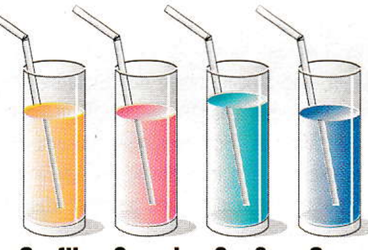
Hersteller: Ocean
Vertrieb: Bomico

Ocean kommt im Tiefflug mit einer handfesten Raumkampfsimulation auf uns zu. „Epic" heißt sie und vereint seriöse Technik-Features mit spaßbringendem Feindbeschuß. Peng Peng! Da es ja primitiv ist, einfach nur irgendwelche Objekte zu zerknallen, gibt es eine akzeptable Hintergrund-Story, die die Wut auf die feindlichen

Raumgleiter schürt. Ein atemberaubender Vorspann stimmt auf den ungleichen Kampf des Spielers ein.

Eigentlich hat aber die Sonne an allem Schuld: ohne ersichtlichen Grund schickt sie sich an, in absehbarer Zeit auseinanderzuplatzen. Da hätten die Menschen schlechte Karten. Glücklicherweise haben die neugierigen Weltraumerforscher bereits den Planeten Ulysses VII entdeckt, auf dem es sich einigermaßen gut leben läßt. Leider ist die





Ersatzwelt ein gutes Stück weit von der Erde entfernt, um genau zu sein: über vier Jahre Flugzeit. Während die langsamen Riesenschiffe die Erdbevölkerung durchs All karren, starten die gelangweilten Rexxonen ihre rasanten Kampfgleiter und gehen auf Jagd - wie manche Neureiche, denen noch ein paar Kerben auf der Elefantenbüchse fehlen. Ganze Flotten der schießwütigen Aliens schwirren aus der Tiefe des Alls herbei.

Tatenlos müssen die Erdlinge nun zusehen, wie sich ein paar heldenhafte Piloten den Angreifern entgegenstellen. Unter ihnen ist auch der Spieler. Er klemmt im Cockpit des irre schnellen Epic-Fighters, der mit seiner raffinierten Waffentechnik praktisch unschlagbar ist. Aber bei so vielen Feinden...? Es gibt nur drei Exemplare dieser Hightech-Flieger. Immerhin muß der Beschützer ganze neun schwierige Missionen im All und auf den Plantenoberflächen durchstehen. Seinen jeweiligen Auftrag erhält der Tollkühne vom Kommandanten der Verteidigungsflotte. Und dann darf simuliert werden. Sehr gute, flüssige Polygonengrafik oszilliert dreidimensional über den Screen. Selbst Einzelheiten sind noch zu erkennen. Ähnlich wie in den üblichen Flugsimulatoren ist das Cockpit des Epic vollgepfropft mit Instrumenten. Anzeigen informieren über Treibstoffvorrat, Tempo und Schäden an Schutzschildern. Feindliche Gleiter werden auf einem Radarschirm als Pünktchen dargestellt. Natürlich kann man auch aus den Fenstern gucken: ein Klick, und man schaut zum Beispiel zur linken Seitenscheibe hinaus. Natürlich kann man nach allen Seiten, auch nach hinten, hinaussehen. Beim Kampf auf direkte Sicht gibt es allerdings ein Problem: die Grafiken der feindlichen Fighter sehen denen der eigenen Kombattanten zum Verwechseln ähnlich. So kommt es zu manchem Eigentor. Bei der radargestützten Zielwahl wird dagegen von der Software ein feindlicher Flieger aufs Korn genommen. Dann aktiviert man eins von acht Waffensystemen.

Besonders praktisch sind die Zielsuchraketen, die ihr Opfer hartnäckig verfolgen. Im Cockpit zeigt ein kleines Fenster, was das Suchauge der Raketenkamera gerade „sieht". Aber auch die unterschiedlichen Laserkanonen heizen den Rexxonen ganz hübsch ein. Anfangs ist es ganz nett, das Zielschießen. Nach einer Weile wünscht man sich Abwechslung und bekommt sie auch! In einer Mission gibt der Epic-Pilot einem irdischen Schiff Geleitschutz, in einer anderen zerbatzt er die rexxonische Radarstation. Mittendrin forscht er auch mal nach einem gigantischen Geschütz, das irgendwo auf einer wüsten Planetenoberfläche steht. Zu einer Art Geschicklichkeitstest wird der turbulente „Ausflug" über ein Minenfeld. In diesem Spiel stellt die Maus ausnahmsweise mal nicht das beste Steuerinstrument dar. Viel verläßlicher geht die Lenkung über die Tastatur von der Hand, weil dann der Gleiter auch dahin flitzt, wo er hin soll. Wenn er mal abstürzt - halb so schlimm, schließlich gibt es noch zwei. Nach dem dritten unfreiwilligen Touch-down ist dann unweigerlich Schluß.

Bei „Epic" handelt es sich um ein feines Spiel mit aufregenden, schwierigen Missionen, mitreißendem Sound und superben Zwischengrafiken. Get it!

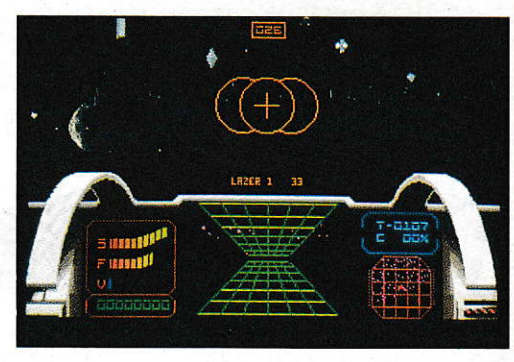
CBO

Hook

Hersteller: Ocean
Vertrieb: Bomico

Wird man jemals richtig erwachsen? Glaubt man der etwas schmalzigen Leinwandstory von „Hook", ist dieser Prozeß zumindest umkehrbar. Erwachsen und seriös geworden ist Peter Pan. Im Jackett, mit Intelligenzbrille auf der Nase, führt er das Leben eines erfolgreichen Geschäftsmannes und Familienvaters. Die Abenteuer seiner Kindheit sind fern, fast schon nicht mehr wahr.

Doch plötzlich wird Peters Welt wieder fantastisch, wenn auch zunächst auf höchst unangenehme Art und Weise: der bärbeißige Piratenhäuptling Hook kidnappt Peters Sprößlinge. Und so wie eine Raupe zur Puppe und







dann zum Schmetterling wird, verwandelt sich der Aktenkoffer-Peter in den Rächer der sieben Meere. Ocean, Spezialist für brutale Ballerspiele, machte mit „Hook“ eine ebenso erfreuliche Wandlung durch und produzierte ein Adventure, vor dem selbst Lucas Arts' Jünger den Hut ziehen müssen. Tropisch bunte Comic-Szenarien mit kuschelweich animierten Sequenzen unterlegen die Handlung. Rasant und nahezu hoppelfrei scrollt der Bildschirm. Berückende und bedrückende Soundeffekte peppen die Atmosphäre auf, stimmungsvolle Musikstücke holen Hollywood auf den Schreibtisch. Angenehm auch, daß „Hook“ polyglott ist: auf Klick zu Spielbeginn werden alle Texte in Deutsch, Englisch oder Französisch angezeigt. Da der Urtext englisch ist, leiden die beiden anderssprachigen Versionen etwas an der Übersetzung. Mancher Witz geht flöten, mancher Satz hakt. Da ließe sich doch wohl was machen - schließlich ist „The Secret of Monkey Island“ auch im Deutschen noch amüsant, verrückt und locker. Dabei ist das Erwachsenenmärchen von Peter Pan an sich wirklich sehr gehaltvoll - ein prachtvolles Thema, eine tolle Herausforderung für die Softwareindustrie.

Trotzdem, das Spiel erzählt die Geschichte nicht so klar und lückenlos wie Spielbergs Kinoerfolg. Manchmal muß man den Film kennen, um ohne lästige Probiererei auf die Lösung zu kommen. Aber schnell muß es gehen, denn Hook hat nach der Entführung ein Ultimatum gestellt. Aus der Märchenwelt tritt die freundliche Fee Tinkerbell an die Seite ihres Schützlings von damals, als Peter in Never Neverland abenteuerte. Der gute Geist ist in Gestalt einer Feuerkugel immer in Peters Nähe. Gemeinsam treffen sie nun in der Freibeuterstadt ein, in der Peter sich ein wenig „overdressed“ fühlt. Da niemand seinen Anzug gegen eine zünftige Piratenkleidung eintauschen will, muß Peter zum Zahnklempner - Goldzähne raus, Gold verkaufen, vernünftiges Abenteurer-Outfit anschaffen. Die modische Ergänzung, ein keckes Hütchen, klaut der Ex-Yuppie. Nun macht aber die verwegene Klamotte aus dem Bürotarzan noch keinen Fantasy-Helden. Peter erfährt dies schmerzhaft, als er mit dem Rauhbein Hook aneinandergerät. Auf das nächste „Rendezvous“ bereitet er sich besser vor. Denn leider kann er nicht mehr fliegen, so wie damals im Neverland. Statt ins städtische Fitness-Center tippelt er durch einen Irrwald zum Trainingslager der Lost Boys. Diese rauhen Brüder möbeln den Schlaffhannes auf, lassen ihn übungshalber pausenlos am Gummiseil aus der Höhe springen. Mit der Zeit erlangt Peter seine alte Form wieder. Seinen Verstand darf er dann an kniffeligen Rätseln wetzen. Quasi zur Belohnung sieht er, wie Hook - als kleines Portrait in der Bildecke dargestellt - sich kopfschüttelnd über jeden Rätselerfolg ärgert. Bei der nächsten Begegnung ergeht es Hook denn auch schlechter als bei der ersten.

Schnell und einfach steuert der Spieler seinen Peterhelden nach Art der Lucas- oder Sierra-Spiele. Mit dem Mauspfeil weist er der Spielfigur den Weg, mit Klicks auf die fünf Aktions-Icons läßt er ihn mit anderen Leuten reden, Dinge nehmen, geben, untersuchen und benutzen. Klickenderweise schnüffelt und forscht man in Gebäuden und packt ins Inventory, was brauchbar erscheint. Auf dem Weg durch die Screens lädt der Computer regelmäßig neue Grafiken nach, natürlich auch dann, wenn Peter in ein Haus hinein oder wieder hinausgeht. Da das Ultimatum läuft, ist Eile angesagt. Also, nicht stundenlang mit anderen Computerbewohnern schwatzen, auch wenn es dank Multiple-Choice-Unterhaltung so prima funktioniert. Leider führt das Ultimatum dazu, daß das Spiel recht bald beendet ist: entweder mit einem Abbruch seitens der Software oder mit Friede, Freude, Eierkuchen und befreiten Kindern. Trotzdem: Ocean hat den Sprung vom dämlichen Bängbäng und blutigen Haudrauf zum beachtlichen Adventure geschafft. Die Umsetzung der Kinogeschichte zum Film ist gut gelungen. Schön gelöst: Peter erfährt jeweils von Fee Tinkerbell, was er als nächstes tun muß. Grafik und Musik stehen den Superadventures der amerikanischen Spezialisten in nichts nach. Auch der Schwierigkeitsgrad und die Komplexität sind in Ordnung. Lediglich ein wenig umfangreicher und witziger könnte „Hook“ sein. Doch Oceans Debüt ist eine Wucht - trotz allem.

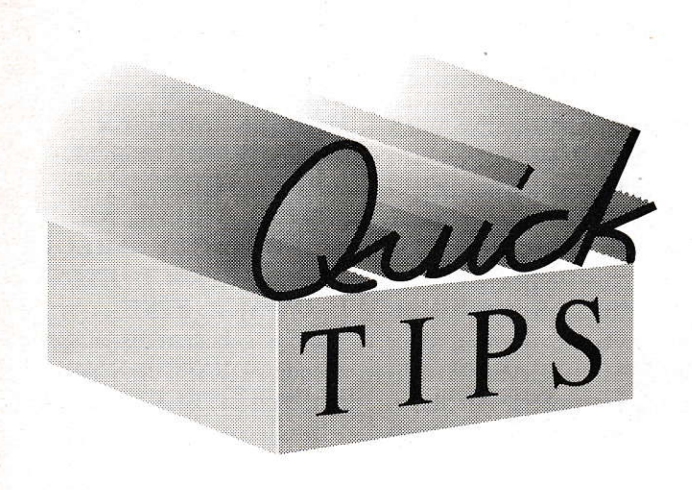
CBO











„BGI spielt falsch“ die zweite...

Dem einen oder anderen, der versucht, mit Pure C auf dem Atari und Turbo C++ unter DOS portable Programme unter Verwendung des BGI zu schreiben, werden schon manche Ungereimtheiten aufgefallen sein. Hier eine weitere Zusammenfassung meiner schlechten Erfahrungen mit Pure C:

Die Funktion *setfillstyle* setzt bei Pure C bei einem Aufruf der Art

setfillstyle(SOLID_FILL, 0)

keine schwarze, sondern eine weiße Füllfarbe, wie bei einem nachfolgenden Aufruf von bar leicht festgestellt werden kann. Bei Turbo C++ wird hingegen - wie bei *setcolor* - eine schwarze Füllfarbe gesetzt.

Die Funktion *putpixel* (siehe Listing) hält sich auch dann an das über *setviewport* angegebene Fenster, wenn das Clipping-Flag beim Aufruf von *setviewport* ausgeschaltet war. Außerdem setzt auch *putpixel* falsche Farben, wenn es im Farbmodus (z. B. mittlere TT-Auflösung) betrieben wird. Vorläufige Abhilfe schafft bei diesem Farbfehler das abgedruckte Programm.

Vorsicht bei Verwendung der Farbe mit der Nummer 15! Viele BGI-Funktionen, so u.a. *setcolor*, übersetzen diese dem VDI als Farbe 16. Diese Farbe existiert aber beim VDI nicht! Zwar habe ich deshalb noch keinen Programmabsturz beobachten können, zweifelhaft ist das Verfahren aber allemal.

Schließlich noch einmal - für alle, die die Ausgabe 1/92 der ST-Computer nicht zur Hand haben - der Fehler bei *clearviewport*: Diese Funktion wartet ab und zu auf einen Tastendruck, was an einem - überflüssigen - Aufruf der VDI-Funktion *vsm_locator* am Ende der Funktion liegt. Man kann den Aufruf durch einen von *vq_mouse* ersetzen, wozu bei dem unten angegebenen Offset in PCBGILIB.LIB das Symbol *vsm_loca* durch *vq_mouse* mit einem Dateimonitor ersetzt werden muß:

PC Version	Offset (hexadezimal)
16. Aug. 1991	$2F17
29. Aug. 1991	$2F29
20. Sep. 1991	$2F29

Ich hoffe, daß dies alle BGI-Fehler sind. Pure Software sollte sie möglichst bald beheben.

R. Lause, Bendorf

```
/* Vorläufiges Beheben des Farbfehlers von */
/* putpixel                                 */
/* Autor: Rüdiger Lause`                    */
/*                                          */
/* (c)1992 by MAXON-Computer                */
/*                                          */

#ifdef __PUREC__
#include <ext.h>    /* für getch     */
#else
#include <conio.h>  /* für getch     */
#endif
#include <graphics.h>

int
    gdriver,     /* BGI-Treiber          */
    gmode,       /* BGI-Mode             */
    maxc;        /* Größte Farbnummer    */

#ifdef __TOS__
/* coltrans bildet die Übersetzungstabelle. */
/* Sie ist global, damit sie nicht bei je-  */
/* dem putpixel-Aufruf initialisiert werden */
/* muß.                                     */
const int coltrans[16] = {
    0,  2,  3,  4,  5,  6,  7,  8,
    9, 10, 11, 12, 13, 14, 15,  1
};
/* Hier die Funktion _putpixel, die die */
/* Farbübersetzung richtigstellt.       */
void _putpixel(int x, int y, int pixelcolor)
{
    if(maxc > 1)
        putpixel(x, y, coltrans[pixelcolor]);
    else
        putpixel(x, y, pixelcolor);
}
/* _putpixel wird für alle putpixel aufgerufen */
#define putpixel(x,y,c) _putpixel(x,y,c)
#endif

/* Ein kleines Beispielprogramm */  43:  int
main(void)
{
    int
        step,
        x, y,
        i,
        col,
        flag;

    /* Initialisierung  */
    initgraph(&gdriver, &gmode, „");
    /* Obligatorisch, da von _putpixel   */
    /* benötigt:                         */
    maxc = getmaxcolor();

    /* Hier das eigentliche Programm     */
    step = (getmaxx()+1) / (getmaxcolor()+1);
    for(flag=0; flag<2; flag++) {
        cleardevice();
        setviewport(step»1, 50,
        getmaxx()-(step»1)+1, getmaxy()-49,
        flag);
        if(step >= 2) {
            col = 0;
            x = -(step»1);
            while(col <= getmaxcolor()) {
                for(y=-50; y<((getmaxy()-99)»1);
                y++)
                    for(i=x; i<x+step; i++)
                        putpixel(i, y, col);
                setfillstyle(SOLID_FILL, col);
                bar(x, y, i-1, getmaxy());
                setcolor(col);
                line(x, -50, i-1, getmaxy());
                x += step;
                col++;
            }
        }
        /* Escape bricht ab!    */
        if(getch()=='\33')
            break;
    }

    /* Grafikende    */
    closegraph();
    /* Tschüß    */
    return 0;
}
```

GFA-Compiler ohne LINEA

Das Grafiksystem der ATARI-Computer beruht bei älteren TOS-Versionen auf den sogenannten LINEA-Codes. Hierbei werden bestimmte Befehlscodes, die die CPU eigentlich gar nicht kennt und die daher zu einem Fehler führen würden, abgefangen und statt dessen verschiedene Routinen zur Grafikausgabe angesprungen. Der Trend geht aber eindeutig weg von den LINEA-Befehlen. Saubere Grafikausgabe per VDI ist gefragt. Leider verwendet GFA-Basic direkt LINEA-Befehle. Sowohl im Interpreter als auch im Compiler sind sie zu finden. Dies kann unter Umständen zu Problemen bei der Grafikausgabe führen. Wie man die unbeliebten LINEA-Befehle aus dem Compiler herauspatcht, soll im Folgenden gezeigt werden. Man kann dazu einen beliebigen Diskettenmonitor benutzen:

Patches:
ASM-Mnemonic = Hex
nop $4E71
rts $4E75

(1) Datei-Offset: $486C
$A000, LA-INIT im Initialisierungsteil Compiler
GFA merkt sich hier A0 und A1.
A0: LA-Variablen
A1: Zeiger auf Tabelle mit Zeigern auf Systemzeichensätze
Patch: nop
VORSICHT: Es werden nach $A000 die aktuellen Inhalte von A0/A1 gerettet, nach dem Patch die Werte nach einem M-SHRINK-Aufruf. Es könnten also beliebige Werte darin stehen, auch Null oder Eins. Dies würde bei manchen GFA-Befehlen dazu führen, daß bei Zugriff auf negative LA-Variablen ein Bus- oder Adreß-Error ausgelöst werden könnte. Deshalb sollte man überhaupt keine Befehle mehr verwenden, die irgendwie nur auf LA-Variablen zugreifen, also z.B. auch kein *SETMOUSE.*

(2) Datei-Offset: $6C82
$A00A, *Hidemouse*, u.a. intern vor VDI-Befehlen
Patch: nop
BESSER: da kein Zugriff auf LA-Variablen Offset $6C6E, direkt am Anfang *Hidemouse*
Patch: rts

(3) Datei-Offset: $6CB6
$A009, *Showmouse*, u.a. intern vor AES-Befehlen
Patch: nop
BESSER: da kein Zugriff auf LA-Variablen Offset $6C94, direkt am Anfang *Showmouse*
Patch: rts

(4) Datei-Offset: $A584, $A58A
Bei Verwendung von VDI-Befehlen macht GFA im Initialisierungsteil ein *vs_clip*, ‚Clipping ein' mit der Bildschirmbreite/-höhe. Beide Werte holt es sich aus Offset -4/-12 der LA-Variablen.
Patches: jeweils $6002 (bra.s relativ zwei Byte weiter).
WICHTIG: !! CLIP OFF am Programmanfang

(5) Datei-Offset: $D55E
$A00A, vor einem *SupExec*, Bedeutung unbekannt.
Patch: nop

Ohne Gewähr auf Vollständigkeit!

Im Interpreter $A009/$A00A *Show/Hide Mouse* rauspatchen! Das gibt zwar leichte „Flecken" beim Bewegen der Maus im Interpreter, aber dafür keine beim laufenden Programm!

Auf eine saubere Schachtelung von *hidems/showms* (über GRAF_MOUSE AES 78) achten! Zu jedem Hide ein Show, sonst gibt's „Flecken". Maus nicht anschalten, wenn sie schon an ist, wie in der Regel nach dem Start von GEM-Programmen, sonst „Flecken".

Folgende Befehle meiden:
**CRSCOL CRSLIN
INPUT MOUSE
MOUSEK MOUSEX
MOUSEY SETMOUSE
SHOWM HIDEM
SPRITE ACHAR ACLIP
ALINE APOLY ARECT
ATEXT BITBLT HLINE
L~A PSET PTST**
...und gegebenenfalls entsprechende VDI/AES-Befehle nutzen! Von BITBLT gibt es eine VDI und eine LA-Variante!

Chr. Conrad, Aachen

```
' (c)1992 by MAXON-Computer
' Autor: Christoph Conrad
' Das Programm durchsucht die GFA-Library
' nach LINEA-Codes und erzeugt eine Liste
' mit den Offsets derselben.
'
DATA  Init, Put Pixel, Get Pixel,
DATA  Line, Horizontal Line
DATA  Filled Rectangle
DATA  Filled Polygon, BitBlt, TextBlt
DATA  Show Mouse, Hide Mouse,
DATA  Transform Mouse, Undraw Sprite
DATA  Draw Sprite, Copy raster form
DATA  Seedfill
DIM la.name$(15)
FOR i&=0 TO 15
  READ la.name$(i&)
NEXT i&
'
lib$="GFA3BLIB"
OPEN "i",#1,lib$
len%=LOF(#1)
CLOSE #1
DIM p|(len%)
ram%=V:p|(0)
BLOAD lib$,ram%
'
OPEN "o",#1,"LINEAADR.DAT"
PRINT #1,"Offsets zum Dateistart (hex)"
'
FOR off%=0 TO len%-1 STEP 2
  w%=CARD{ram%+off%}
  IF &HA000<=w% AND w%<=&HA00F
    info$=HEX$(off%,6)+" : "+HEX$(w%,4)+
    la.name$(w% AND &HF)
    PRINT info$
    PRINT #1;info$
  ENDIF
NEXT off%
CLOSE #1
```

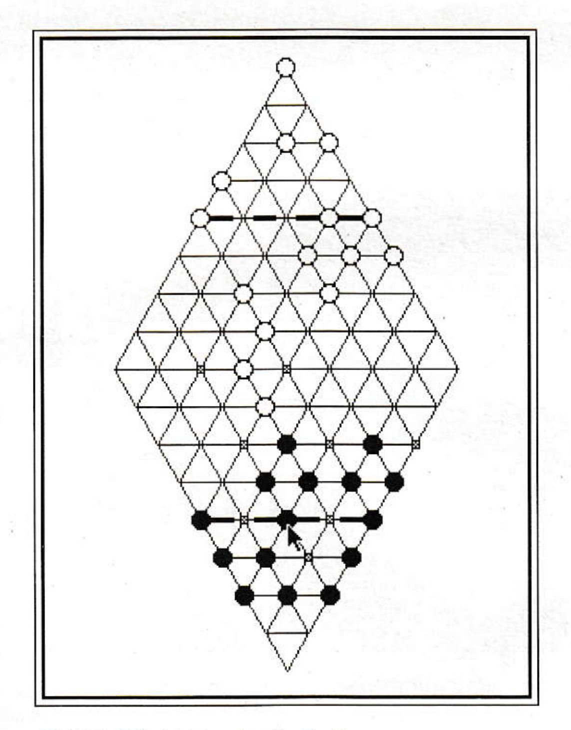




Sie sind beliebt bei jung und alt. Sie verbreiten überall gute Laune. Sie sind innovativ, kreativ und überdies völlig uneigennützig. Und sie erscheinen jeden Monat in der PD-NEWS – die besten Programme aus der PD-Serie der ST-Computer.

Halma-Partner

Eigentlich ist es ja schöner, die klassischen Brettspiele auch gegen Partner aus Fleisch und Blut zu spielen, aber manchmal hat man keinen Spielpartner, oder es reizt einfach, gegen die unbestechliche Logik des Computers anzutreten. Aus der Sicht eines mathematisch interessierten Programmierers ist sicherlich die Entwicklung eines Spielalgorithmus' eine lohnende Herausforderung. - Wie dem auch sei, jetzt kann man auch Halma gegen der Atari spielen (ST-PD # 493). Das von Joachim Meyer in der Programmiersprache C entwickelte Programm präsentiert sich schlicht und ohne Spielerei in einem GEM-Fenster. Die Regeln von Halma werden als bekannt vorausgesetzt. Der Spieler bekommt die schwarzen Figuren und darf außerdem den ersten Zug machen. Dazu klickt man zunächst den Spielstein an, den man setzen möchte und anschließend die gewünschte Zielposition. Hat man sich entschieden, hüpft der Stein wie gewohnt über das Spielbrett. Dabei zeigt das Programm automatisch alle er-laubten Züge an. Das kann von Vorteil sein, da man so auf Möglichkeiten aufmerksam wird, die man sonst übersehen hätte, verführt aber ebenfalls dazu, die eigene Taktik aus den Augen zu verlieren.

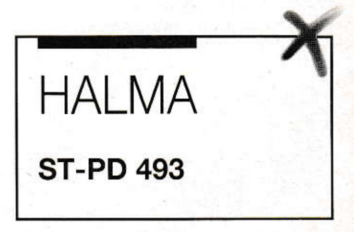


HALMA läßt sich komfortabel bedienen.

Neben dem Spielfeld gibt der Computerspieler Insiderinformationen und in der Schlußphase Zugabsichten an. Bei der Spielstärke „schwach“ denkt er dann zwei, bei „stark“ drei Züge voraus. Außerdem kann man beliebig viele Züge zurücknehmen und wieder vorwärts gehen sowie die gesamte Partie abspeichern.

Einziger Kritikpunkt an HALMA ist die Größe des Brettes. Durch die aufrechte Darstellung ist es auf dem Monitor doch ziemlich klein ausgefallen, so daß man schon sehr genau hinschauen und klicken muß, wenn man ziehen möchte. Ein um 90° gedrehtes Brett sollte wenigstens als Alternative zur Verfügung stehen. Dafür ist die Spielstärke des Programmes beachtlich. Es dauert schon einige Zeit, bis man sich als Normalverbraucher einmal gegen die Maschine durchsetzen kann. Leider kann man sich dann nicht wie sonst üblich in eine Siegerliste eintragen.

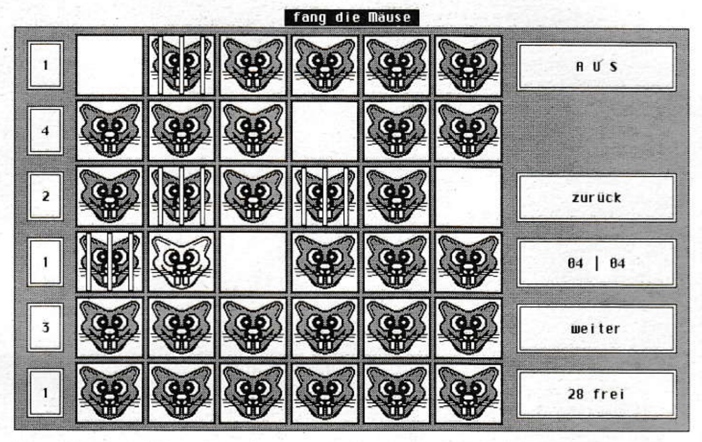
thl



Mäuseplage

Das Spiel TI_MÄUSE (ST-PD #498) von Rudolf Tiemann ist gewissermaßen ein Upgrade von RATTE (ST-PD #232). Im Gegensatz zum Vorgänger läuft es jetzt in Schwarzweiß und Farbe und verträgt sich mit allen Betriebssystemversionen. Darüber hinaus liegt der Quell-Code in GFA-BASIC dem Programm bei.

Die Anleitung für dieses Denkspiel ist in das Programm eingebaut: es gilt, eine Mäuseplage einzudämmen. Auf einem sechs mal sechs Kästchen großen Feld tummeln sich bis zu 34 Mäuse; zwei, drei oder vier Felder sind unbesetzt. Ziel des Spieles ist es, die Mäuse hinter Gitter zu bringen, in dem man sie auf die leeren Felder scheucht, die nämlich Mausefallen sind. Das hört sich einfachen an, als es ist, denn die Tierchen können sich nur über eine bestimmte Anzahl von Feldern bewegen. Wieweit sie springen können, steht vor der entsprechenden Zeile. Man muß also darauf achten, daß die enstehende Lücke durch eine andere Maus erreicht werden kann, weil man sonst irgendwann keine Maus mehr bewegen kann, aber trotzdem noch nicht alle gefangen sind. Als einzige Hilfe gibt es die Möglichkeit, Züge zurückzunehmen. Sonst hilft nur vorausschauendes Denken.



Vier Mäuse sind bereits gefangen, die fünfte ist gleich fällig.

Dem Programm liegt zwar ein sehr einfaches Spielprinzip zugrunde, das einen aber auf lange Zeit fesseln kann, sofern man etwas für Knobeleien übrig hat. Im Gegensatz zu den TI_FROGS (ebenfalls ST-PD #498) ist es kein Kinderspiel, bei dem man die Lösung einfach aus dem Ärmel schütteln kann.

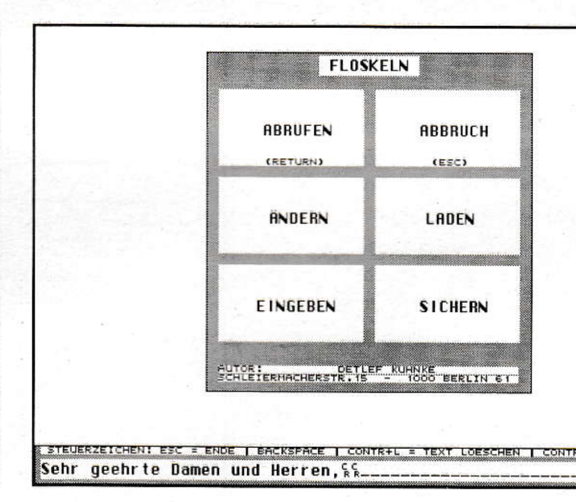
thl



Tipphilfe

Nicht selten kommt es vor, daß man immer wieder die gleichen Texte eintippt: Anschriften, Absender, Anreden, Typenbezeichnungen usw. Das ist nicht nur langweilig, sondern häufig schleichen sich dabei außerdem Tippfehler ein. Einige Programme bieten über Makrorekorder oder spezielle Floskeltasten Abhilfe, bei anderen muß man sich etwas einfallen lassen.

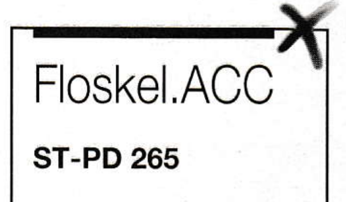
Eine Möglichkeit diesem Mangel abzuhelfen, besteht mit dem FLOSKEL-Accessory von Detlef Kuhnke (ST-PD #265). Dieses kleine Programm speichert 17 bis zu 63 Zeichen lange Texte und fügt sie auf Tastendruck in den aktuellen Text einer Textverarbeitung, eines Editors oder einer Tabellenkalkulation ein. Dabei können nicht nur ASCII-, sondern auch Steuerzeichen verwendet werden. Selbstverständlich können die Texte auch gespeichert oder nachgeladen werden. In einer kleinen Tabelle werden die Texte bzw. deren Anfänge mit der dazugehörenden Taste aufgelistet.



Die Floskeln lassen sich per Tastendruck auswählen.

Der große Vorteil dieser Lösung ist, daß diese Makros nicht an ein bestimmtes Programm gebunden sind, sondern - theoretisch - in allen Anwendungen, die Accessories zulassen, in gleicher Weise zur Verfügung stehen. Der Haken der Sache ist aber, daß einige Programme den Text nicht übernehmen oder die Steuerzeichen ignorieren. Da helfen nur Probieren und ein bißchen Glück. Es ist aber sicherlich einen Versuch wert, dies genauer zu ergründen.

thl





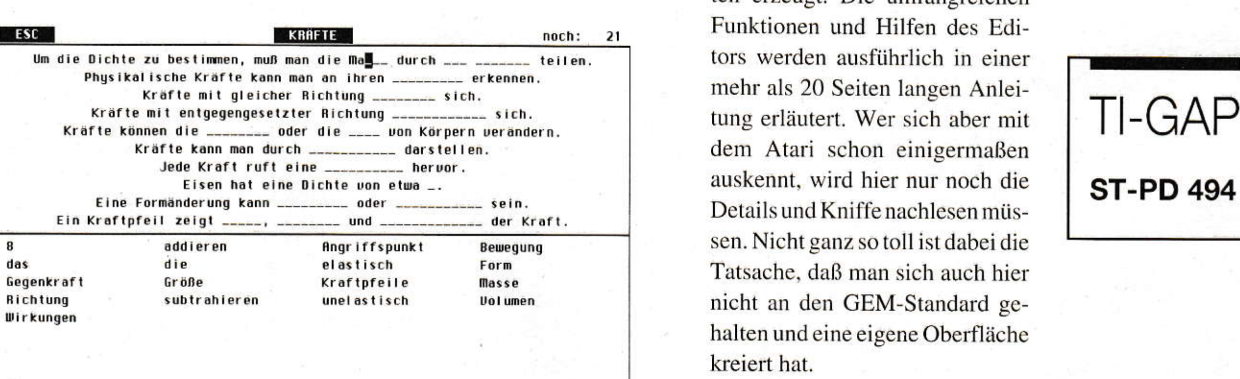
Wissenslücken schließen

Immer häufiger hält der Computer Einzug in die Schule oder wird wenigstens indirekt dafür eingesetzt (vgl. Computerprojekt, ST Computer 10/1991). Neben einem ganzen Batzen Spielen (ST-PD #498) hat Rudolf Tiemann auch ein Lückentext-Programm TI_GAP für den schulischen Einsatz konzipiert (ST-PD #494). Das in GFA-BASIC 3.0 entwickelte Programm arbeitet sowohl in der hohen wie auch der mittleren Auflösung, wobei natürlich einem SM 124/144 der Vorzug zu geben ist.

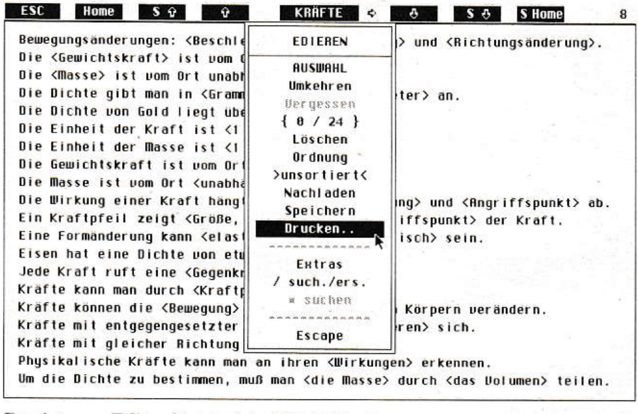
Sicherlich kann ein Computerprogramm kaum einen guten Lehrer ersetzen, aber zum Faktenlernen ist er in der Regel gut zu gebrauchen und sehr geduldig. Der Lückentext bietet die Möglichkeit, auch Wissen abzufragen, das nicht in strengen Wortpaaren wie z. B. Vokabeln vorliegt. Für den Lernenden - der ja nicht unbegingt Computerfreak sein muß - ist die Bedienung des Programmes ganz einfach: Im Abfragemodus wird ein Textausschnitt von maximal zehn Zeilen angezeigt, bei dem die Lücken durch den allseits bekannten Unterstrich hervorgehoben sind. Als Lösungshilfe werden im unteren Teil des Bildschirmes die fehlenden Begriffe in alphabetischer Reihenfolge aufgeführt.

Normalerweise beginnt man bei der ersten Lücke, kann aber per Maus zu einer anderen Stelle hüpfen. Die Antworten sind unter Berücksichtigung der Groß- und Kleinschreibung einzugeben. Wird ein falscher Buchstabe getippt, ertönt ein Signal, bis man den richtigen eingibt. Jede richtig gelöste Zeile verschwindet und wird durch eine neue ersetzt, bis die Lektion abgeschlossen ist. Wer trotz der Unterstriche (sie entsprechen ja der Buchstabenanzahl des Lösungswortes) und der Wortliste nicht klarkommt, kann per „Escape" abbrechen. Eine Hilfe, die den Anfangsbuchstaben verrät, gibt es leider nicht.

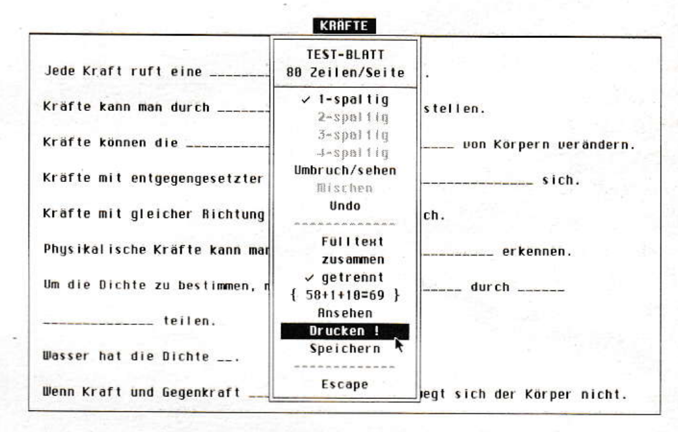
Obwohl es sich bei den Lektionen - d. h. den einzelnen Lückentexten - um einfache ASCII-Dateien handelt, ist ein Editor in das Programm integriert. Er hat zwar nicht alle Funktionen, die ein normaler Editor hat, bietet dafür aber einige Hilfen zur Erstellung dieser speziellen Texte, was wichtiger und sinnvoller ist. So setzt er per Tastendruck die Markierungen für die Wörter, die erfragt werden sollen, oder bietet eine pfiffige Suchen-und-ersetzen-Funktion: ist das gefundene Wort klein geschrieben, wird das neue ebenfalls klein geschrieben, beginnt das gefunde Wort groß, wird der erste Buchstabe des Ersatz-Strings auch groß geschrieben. Darüber hinaus lassen sich Lektionen zusammenfügen, mixen, sortieren und doppelte Zeilen automatisch löschen. Die Länge einer Lektion ist auf 1000



Der Abfragemodus



Der interne Editor bietet viele Möglichkeiten.



Die Lektionen lassen sich auch ausdrucken.

Zeilen begrenzt, wobei es wahrscheinlich sowieso sinnvoller ist, die Texte nicht so lang zu machen und lieber auf mehrere Dateien zu verteilen.

Da der Computer trotz des Preisverfalls noch immer ein recht kostspieliges Arbeitsgerät ist, lassen sich die ausgetüftelten Lückentexte auch zu Papier bringen. Um sie per Hand ausfüllen zu können, werden der Zeilenabstand sowie die Lücken automatisch vergrößert. Sind die einzelnen Zeilen sehr kurz, werden bis zu vier Textspalten erzeugt. Die umfangreichen Funktionen und Hilfen des Editors werden ausführlich in einer mehr als 20 Seiten langen Anleitung erläutert. Wer sich aber mit dem Atari schon einigermaßen auskennt, wird hier nur noch die Details und Kniffe nachlesen müssen. Nicht ganz so toll ist dabei die Tatsache, daß man sich auch hier nicht an den GEM-Standard gehalten und eine eigene Oberfläche kreiert hat.

Wer Zweifel an der Flexibilität eines solchen Programmes hegt, sei auf die umfangreichen Beispieldateien verwiesen: nicht nur „richtige" Texte, sondern auch Matheaufgaben lassen sich auf diese Weise abfragen. Neben unheimlich vielen Sprichwörtern gibt es auch Fragen zur Biologie, Grammatik, Physik und Mathematik. Optimal einsetzen läßt sich das Programm natürlich nur, wenn man es an die individuellen Erfordernisse anpaßt. Vor allem muß man darauf achten, daß der Text nicht durch zu viele Lücken so zerissen wird, daß man wirklich nur noch raten kann.

thl

TI-GAP

ST-PD 494

Immer up to date

	Programmname	Version	Daten
☛	ACS	1.05	N HML
	AHDI	5.00	N HML
	Arabesque	1.14	N H
	Arabesque Professional	2.14	N H
	Argon	1.42	N
	Augur OCR	1.6	J HML
	Avant trace	1.0	
	Avant vektor	1.2	
	Avant plot	1.2	
	Banktransfer	1.0	N H
	Barcode ST	1.0	N HM
	Basichart	1.21	N HM
	BASIC-Konverter nach C	3.1	N H
	BTX-Börsenmanager	4.0	N H
	BTX/VTX-Manager	4.0	N H L
	Calamus 1.09N	18.07.91	N H 1M
	Calamus S	06.05.92	N H F 1M
	Calamus SL	15.04.92	N H F 1M
☛	Calligrapher Professional	2.28	N HMLF 1M
	Cashflow	1.0	N H 1M
	Charly Image	1.03	N HML
	Chemograph Plus	4.31	
	Chemplot	2.0c	
	Chemotech	1.2	
	CIS-L&G	2.1	N H 2M
	Convector Zwei	1.00	N H
	ConVert	2.02	N H 1M
	Crypton	1.8	N
	Cubase	3.0	
	CW-Chart	8.0	N H 1M
	CyPress	1.11	N H
	dBMAN	5.3	N HML
☛	Didot Professional s/w	3.142	
☛	Didot Professional Color	4.142	
	Diskus	2.50	N HM F 1M
	Easybase	1.39	N HM
	easyHeadline	1.0	N H F 1M
	Easy Rider Assembler	3.00	N HML
	Easy Rider Assembler 030	3.52	N HML
	Easy Rider Reassembler	3.06	N HML
	Easy Rider Reassembler 030	3.54	N HML
	Edison	1.00	N HM
	1stEuroTrenn	1.0	N H F 1M
	fibuMAN	4.0	N H
	fibuSTAT	2.3	N H
	Flexdisk	1.6	N HML
	FM-Meßtechnik	1.0.b	N HM
	FontMaker	1.2	N H 1M
	Formula	1.0	N H 1M
	Formular plus	3.07	N H F 1M
	FTL Modula-2	1.18	N HM
	Gemini	1.21	N HML
	GEMinterface ST	1.1.	N HML
	GFA-BASIC-Compiler	3.6	N HML
	GFA-BASIC-Interpreter	3.6	N HML
	Hänisch Modula-2	4.0	N HML
	H.Modula-2-Runtime-Debugger	2.0	N HML
	H.Modula-2-ONYX-Assembler	1.72	N HML
	H.Modula-2-Window-Library	4.55	N HML
	H.Modula-2-GEMplus-Library	2.5	N HML
	Hard Disk Accelerator	1.0	N HML
	Hard Disk Sentry	1.10	
	Hard Disk Toolkit	2.0	N HM
	Harddisk Utility	3.0	N HM
	Harlekin	2.06	N HM
	HAROFAKT	8.9	N H 1M
	HEADline	4.0	N H F 1M
	Illuminator	2.0	N HML
	ISI-Interpreter	1.20	N HM
	Interface	1.07	N HML
	Junior Office	1.3	N H F 1M
	Junior Prommer	2.33	N HM
	Kleisterscheibe	2.32	N HM
	Kobold	1.07	N H
	K-Resource	2.0	N HM
	K-Spread 4	4.19	N HML
	1st_Lektor	1.2	N HM
	Lektorat	1.2	N HML
	Lattice C	5.51	N HML
	Maxon PASCAL	1.5	N HML
	Maxon PROLOG	1.1	N H 1M
	Maxidat+	3.02	N HM F
	Means V2	2.0	J H 1M
	Mega Paint II Professional	3.01	N H 1M
	Megamax Modula 2	3.5	N HM
	MGP GAL-Prommer	2.1	N H
	Mortimer Plus	1.12	
	MPe II plus	1.02	N H 1M
	MT C-Shell	1.2	N HM 1M
	MultiGEM	2.0	N HML
	MultiTerm-pro	1.4	N HM 1M
	NeoDesk	3.0	N HML
	Notator	3.1	
	NVDI	2.03	N HML
	Omikron BASIC-Compiler	3.57	N HML
	Omikron BASIC 68881-Compiler	3.06	N HML
	Omikron BASIC Interpreter	3.57	N HML
	Omikron.BASIC TT	4.07	N HML
	Outline-Art	1.0	N H 1M
	Outside	1.03	N HML
	PAM's TERM/4014	3.012e	N H
	PAM's TurboDisk	1.7	N HML
	PAM's NET	1.2	N HML
	PCB-layout	1.33	N H
	PCB-layout plus	1.13	N H
	PegaDress	1.0	N H
	PegaFakt	2.0	N H
	PegaStic	1.1	N H
	Phoenix	2.0	N HML
	phs-BTX-Box	6.1	N HML 1M
	phs-ST-Box	1.2	N HM
	phs-Boxtalk	1.0	N HM 1M
	phs-Boxedi	1.0	N HML 1M
	Platon	2.01	N H 1M
	1st Proportional	3.13	N HM
	Prospero Pascal	2.153	N HML
	Prospero Fortran	2.153	N HML
	Prospero C-Compiler	1.144	N HML
	Prospero Developers Toolkit	1.111	N HML
	Protos	1.1	N H 1M
	Pure C	1.0	N HML
	Querdruck2	2.10	N HM
	QFax/Pro	3.22	N HML
	ReProk international	2.03	N HM 1M
	Repro Studio junior	2.0	
	Repro Studio ST	2.0	
	Repro Studio pro	1.1	
☛	Retouche	1.11	
☛	Retouche Professional	1.30	
☛	Retouche Professional CD	1.04	
	rho-Data	2.45	H F 1M
	rho-Datenlogger II	7.02	H F 2M
	rho-Prozess	1.16	H F 2M
	rho-Transient plus	3.05	H F 1M
	Rufus	1.10	N HML 1M
	Scansoft	3.2	N HML
	Scarabus	3.0	N H
	Script	2.2	N HM
	SDOindex	1.5	N H 1M
	SDOgraph	2.2	N H 1M
	SDOmerge	1.5	N H 1M
	SDOpreview	3.01	N H 1M
☛	Sherlook	2.42	
☛	Sherlook Professional	3.4	
	Signum! 3	1.0	N HM 2M
	Simula	3.0	N HML 1M
☛	Skyplot plus	5.2	N H F 1M
	SPC-Modula-2	2.0	N HML
	Spectre	3.0	J HM
	SPS ST	1.5	N H 1M
	STAD	1.3+	N H
	Steuer-Tax 2.91	1.11	N HM
	Steuer-Tax 3.91	1.11	N HM
	Steve	3.0	N H
	SparrowText	2.28	N H 1M
	ST-Fakt	3.24	N H 1M
	ST-Fibu	3.14	N H 1M
	ST-Giro	1.15	N H 1M
	ST Pascal plus	2.10	N HM
	Syntex OCR	1.2	N HML
	Technobox Drafter/2	2.0	J H 1M
	Technobox CAD/2-ST/TT	1.4	J H 2M
	Tele Office	1.3	N H F 1M
	Tempus Editor	2.12	N HM
	Tempus Word	2.0	N H 1M
	That's Write	2.0	N HM
	Themadat	4.12	
	TIM II	1.0	N H 1M
☛	tms Vektor	3.2	N H F 4M
☛	tms Vektor Bambino	1.2	N H F 1M
☛	tms Cranach Paint	1.2	N H F 1M
☛	tms Cranach Studio	1.2	N H F 4M
	Typeart	1.04	N H F 1M
	UIS II + Hermes	2.5	
☛	Ultimade-PCB	2.0	N H 1M
	VecToMap	2.55	N H 1M
	Writer ST	2.0	N HM
☛	Wordflair II	1.07	
	Wordplus	3.20	N HML
☛	X-Act	3.0	N HM F

Irrtum vorbehalten! Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung, L = niedrige Auflösung, F = Farbe ab 640x480, 1M = mindestens 1 Megabyte, ☛ = Änderung gegenüber letzter Ausgabe

In der nächsten ST-Computer lesen Sie unter anderem

ATARI-Messe 1992 in Düsseldorf

Und wieder ist es soweit: in Düsseldorf flattern die ATARI-Fahnen. Natürlich sind auch wir wieder mit dabei und berichten ausführlich über Trends, Neuheiten und Gespräche von der größten ATARI-Show der Welt. Lassen Sie sich überraschen.

High-End-Festplatten

Massenspeicher werden immer kleiner, deren Kapazitäten verhalten sich aber genau umgekehrt. 3,5"-Festplatten mit mehr als 500 Megabyte Speicherplatz sind keine Seltenheit mehr. Grund genug, die „Mega-Zwerge" einmal genauer zu betrachten. Wir testen anschlußfertige Festplattensysteme für ST-/STE- und TT-Computer ab 200 Megabyte aufwärts. Was können diese modernen Geräte wirklich leisten? Lohnt sich eine solche Anschaffung?

Inshape - ein Profi-Raytracer

Raytracing war bislang eine Domäne von PCs, Amigas und Apples. ATARI-Computer hatten auf diesem Gebiet noch nicht so viel zu bieten. Das soll sich nun ändern: mit Inshape kommt erstmalig ein professioneller Raytracer für den ATARI-TT auf den Markt, der insbesondere die Hardware-Eigenschaften dieses Rechengiganten ausnutzt. Erfahren Sie in unserem Testbericht, ob Inshape gegen die Konkurrenten der anderen Computer-Systeme bestehen kann.

Eine Pflanze läßt grüßen - Papyrus

Eine neue Textverarbeitung reiht sich in die Riege der so zahlreich vertretenen Mitbewerber ein. Papyrus soll alles bieten, was man sich von einer professionellen Textverarbeitung erhofft. Unser Spezialist für DTP und Textverarbeitung hat sich mit dem Programm auseinandergesetzt. Lesen Sie seinen Test.

Die nächste ST-Computer erscheint am Fr., den 25.09.1992

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir weiterhin vorhaben, die Qualität zu steigern, haben wir Redakteure eine große Bitte an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur **donnerstags von 14[00]-17[00] Uhr** unter der Rufnummer 06196/481814 telefonisch beantwortet werden können.

Außerdem besteht die Möglichkeit, Fragen in die MAXON-Mailbox per Modem zu senden. Die Box ist unter der Nummer 06196/43780 wochentags von 20[00] Uhr abends bis 8[00] Uhr morgens mit den Parametern 8N1 zu erreichen.

Natürlich können wir Ihnen **keine** speziellen Einkaufstips geben. Wenden Sie sich in diesem Fall bitte an einen Fachhändler. Wir können nur Fragen zur ST-Computer beantworten.

Vielen Dank für Ihr Verständnis!

Impressum ST Computer

Chefredakteur; Harald Egel (HE)

Redaktion:
Harald Egel (HE), Dieter Kühner (DK), Joachim Merz (JM), Christian Möller (CM)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO), Claus Brod (CB), Ingo Brümmer (IB), Derek dela Fuente (ddf), U.Seimet (US), R.Tolksdorf (RT), Thomas Werner (TW)

Autoren dieser Ausgabe:
F.Baumgart, V.Brixius, R.Esser, A.Lauterbach, T.Luhm (thl), G.Michel, R.Nitz, R.Osten, J.Piscol, D.Rabich, M.Regitz, A.Rosemeier, L.Schwabe, H.Simon, M.Steinle, G.Süßmuth, H.Wörndl-Aichriedler

Auslandskorrespondenz:
D.Dela Fuente (UK)

Redaktion: MAXON Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 14, FAX : 0 61 96/4 11 37

Verlag: Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57, FAX : 0 61 51/59 10 47 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Sterna, H. Arbogast

Anzeigenpreise:
nach Preisliste Nr.7, gültig ab 2.1.92
ISSN 0932-0385

Grafische Gestaltung:
Manfred Zimmermann, Raoul Deubler

Titelgestaltung:
Axel Weigand

Fotografie:
Christian Möller

Illustration:
Manfred Zimmermann

Produktion:
B.Kissner

Druck:
Frotscher Druck GmbH

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 8,-, ÖS 64,-, SFr 8,-
Jahresabonnement: DM 80,-
Europ. Ausland : DM 100,- Luftpost : DM 130,-
In den Preisen sind die gesetzliche MWSt. und die Zustellgebühren enthalten.

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern der MAXON Computer GmbH. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

DIGITAL DATA DEICKE, NORDRING 9, 3000 HANNOVER 1, TEL: 0511/6370 -54, -55

* =Mehr Festplatte für's gleiche Geld: ddd-Compress. Beim Mega STE, TT und der MicroDisk im Preis enthalten !

↓ = Preisrutsch bei ddd. - Sind Sie dabei ?

Zubehör

Ganzseitenmonitor mit Farbgrafikkarte VME für Mega STE/TT	1888,- ↓
Laserdrucker für ATARI ab	1694,-
Cherry Tastatur für ST, STE , TT	249,-
1 MByte SIMM für alle STE	66,- ↓
Coprozessor für Mega STE	99,-
Coprozessor für Mega ST	299,-
SM 146 für ST, STE	294,-
ddd-Compress für ST, STE, TT	88,-
Leiser Lüfter für Mega ST	39,-
TOS 2.06 für STE	99,- ↓
TOS 2.06 + Bridge	133,- ↓
SCSI-Contr. + Deckel f. STE	144,- ↓
Eizo F550i, 17" Farbmonitor	2498,-

TT

Preisrutsch auch beim TT !

TT 030-4MB	2444,- ↓
TT 030-8MB	a.A. ↓
TT 030-20MB	3888,- ↓

Atari TT jetzt im Angebot. Lieferbar ! Mit HD-Laufwerk (720KB und 1.44MB)

Festplattenaufpreise:
von 48/74MB* 366,-
bis 425/650MB* 2198,-

Rechenleistung pur: der TT

Mega STE

Unsere neuen Preise:

Mega STE 1	DM 894,- ↓
Mega STE 2	DM 1030,- ↓
Mega STE 4	DM 1160,- ↓
Aufpreis für HD-Laufwerk	DM +99,-

Festplatten (eingebaut):

mit 48/74MB*	+388,- ↓
mit 52/80MB*	+533,- ↓
mit 85/130MB*	a.A. ↓
mit 105/160MB*	+799,- ↓
mit 210/320MB*	+1398,- ↓
mit 425/650MB*	+2222,-

Warum kaufen so viele Ihren Mega STE bei uns ? Dafür gibt es einige Gründe:

- NEU: Mehr Speicherkapazität fürs gleiche Geld. Ab sofort liefern wir alle Festplatten mit ddd-Compress (mehr als 50% höhere Kapazität) ohne Aufpreis aus.
- Wir liefern alle Mega STEs, die mit Festplatte ausgerüstet werden, komplett eingerichtet aus.
- Wir liefern zusätzlich den von der MicroDisk bekannten Treiber mit (Auto-Shutdown, 2-fache Bootwahl, mit Cache bis zu 3.5 mal schneller usw.).
- Wir liefern alle Mega STEs auch mit eingebautem HD-Laufwerk, dem kommenden Standard, aus.
- Wir liefern auf Wunsch statt des ATARI-Controllers (max. 1 Platte) auch den schnellen, von der MicroDisk bekannten CAT-Controller (+66,-), an den bis zu 7 Festplatten angeschlossen werden können. Optional ist ein SCSI-Ausgang erhältlich.
- Wir sind bastlerfreundlich, d.h. zusätzlich von Ihnen eingebaute Teile (z.B. eine Grafikkarte o.ä.) führen nicht automatisch zum Garantieverlust.
- Wir liefern im Paket zusammen mit dem Mega STE, z.B. den SM 146 (+277,-), Coprozessor (+99,-), Cherry Tastatur (+188,-), Calamus (+294,-), den Laserdrucker (+1799,-) Ganzseitenmonitor (+1798,-) überaus günstig.
- Wir sind für Sie da und können aufgrund unserer Erfahrung als Entwickler und Hersteller auch bei schwierigen Fragen helfen.

HD-Diskettenstationen

HD-Diskettenstationen (720KB **und** 1.44MB): der Standard von morgen. Jetzt aufsteigen. Ausführliches Info kostenlos anfordern.

3.5" HD-Station zum Einbau incl. Anleitung	DM 129,-
3.5" wie vor mit ddd HD-Modul	DM 185,-
3.5" externe HD-Station anschlußfertig	DM 196,-
3.5" wie vor mit ddd HD-Modul	DM 255,-
5.25" HD-Station zum "Einbau" incl. Anleitung	DM 149,-
5.25" wie vor mit ddd HD-Modul	DM 199,-
5.25" externe HD-Station anschlußfertig	DM 222,-
5.25" wie vor mit ddd HD-Modul	DM 277,-

Special: zwei Stationen zum Preis einer einzelnen ! Für DM 249,- bekommen Sie eine intern einzubauende 3.5" HD-Station, ein HD-Modul und alle Teile, um mit Ihrem alten Laufwerk ein externes 3.5" Laufwerk aufzubauen. Für DM 50,- mehr bauen wir alles für Sie professionell ein und um. Ist das was ?

ddd-MicroDisk

Die ddd MicroDisk ist eine sehr kleine anschlußfertige externe Festplatte (siehe Bild, Abbildung 1:1). Bei der Entwicklung dieser Festplattengeneration wurde besonders auf hohe Zuverlässigkeit und lange Lebensdauer Wert gelegt. So verwenden wir z.B. längsgeregelte Netzteile (eingebaut) anstatt anfälliger Schaltnetzteile, erreichen durch gute Kühlung gerade 25 Grad Celsius Laufwerkstemperatur (entscheidend für Datensicherheit und Lebensdauer) anstatt 40 oder gar 60 Grad, verwenden kugelgelagerte Lüfter für leisen Lauf und lange Lebensdauer, verwenden VDE-gerechte Bauteile zu Ihrer Sicherheit, puffern DMA-In und OUT, haben den SCSI-Bus herausgeführt und benutzen einen der schnellsten Controller. NEU: Ab sofort liefern wir alle MicroDisk-Festplatten mit ddd-Compress für ca. 50% mehr Speichervermögen aus. Mehr Kapazität zum gleichen Preis!

Abbildung in Originalgröße

Der Controller
Speziell für höchste Geschwindigkeit entwickelt, garantieren wir einen Interleave von 1 und erreichen Übertragungsraten bis über 1500 KByte/s. Integrierter Hardwareschreibschutz zur Sicherheit vor Viren (vergessen Sie Passwörter !). Bis zu 7 Festplatten anschließbar. Adressen von außen bzw. durch Software einstellbar (s.Software) Echtzeituhr (baugleich dem Mega ST) nachrüstbar.

Die Software
Der Treiber ist voll Atari AHDI 4.0 kompatibel. Neu: Durch Cache bis 512KB (einstellbar) um bis zu Faktor 3.4 schneller ! Neu: Softwaremäßige Unit-Adresseinstellung (52er u. 105er). Voll autobootfähig von jeder Partition. Jede MicroDisk wird komplett eingerichtet geliefert, also anschließen, einschalten und sofort arbeiten (wie mit Disketten, nur bis zu 50 mal schneller).

Die Laufwerke
Zum Einsatz kommen ausschließlich modernste 3.5" SCSI-Drives von Seagate und Quantum. Aber Achtung: Quantum ist nicht gleich Quantum ! Wir verwenden nur die Laufwerke der neuen LPS Serie mit 1" Bauhöhe aufgrund des geringeren Laufgeräusches und der höheren Geschwindigkeit. Alle Laufwerke haben Hardware-Autopark-Funktion, parken überflüssig.

Die Preise
ddd-MicroDisk 48/74* mit Seagate ST 157N-1 **DM 694,-** ↓ (555,-)
ddd-MicroDisk 52/80* mit Quantum LPS 52 S **DM 833,-** ↓ (666,-)
ddd-MicroDisk 105/160* mit Quantum LPS 105 S **DM 1111,-** ↓ (944,-)
ddd-MicroDisk 210/320* mit Rodime 3259 **DM 1666,-** ↓ (1444,-)
ddd-MicroDisk 425/650* mit Fujitsu-Laufwerk **DM 2444,-** ↓ (2222,-)

Kit-Preise in Klammer (Platte, Controller, Kabel, Software)

Die Cherry G80/1000 Tastatur anschlußfertig(!!!) für alle ST,STE u TT gibt's bei uns für nur DM 249,-

Versand per NN, europaweit und Direktverkauf in Hannover

Öffnungszeiten: MO. - FR. von 10 - 18 Uhr durchgehend
Samstag und Sonntag geschlossen.

Es gelten unsere Geschäftsbedingungen

Rufen sie doch mal an

digital data deicke, einer der weltweit größten Anbieter im ST-Bereich rät: Achten Sie auf guten technischen Support und eine erreichbare Hotline. Es zahlt sich aus !